Mitigating Cross-Domain SEU Corruption in FPGA-Based AI Accelerators for Space Applications

Mitigating Cross-Domain SEU Corruption in FPGA-Based AI Accelerators for Space Applications

Invited Paper

Sarah Azimi
Politecnico di Torino
Turin, Italy
sarah.azimi@polito.it

Eleonora Vacca
Politecnico di Torino
Turin, Italy
eleonora.vacca@polito.it

Corrado De Sio
Politecnico di Torino
Turin, Italy
corrado.desio@polito.it

Luca Sterpone
Politecnico di Torino
Turin, Italy
luca.sterpone@polito.it

Andrea Portaluri
Politecnico di Torino
Turin, Italy
andrea.portaluri@polito.it

Abstract

SRAM-based FPGAs have become increasingly popular for mission-critical applications due to their high computational performance and reconfigurability, particularly in AI-driven workloads. However, their vulnerability to radiation-induced faults, especially Single Event Upsets (SEUs), demands robust mitigation strategies. While techniques like modular redundancy offer protection, they remain susceptible to cross-domain errors that can simultaneously affect multiple cores.

This paper explores the impact of placement-level isolation as a mitigation strategy for SEUs in multi-core FPGA designs. We propose and evaluate three layout configurations with varying levels of resource isolation, implemented on a Xilinx UltraScale+ device. The designs were subjected to high-energy proton irradiation to assess their fault resilience. Results show that custom placement strategies—beyond default vendor tools—can improve resistance to cross-domain soft errors by up to 36%, offering a promising complement to traditional fault-tolerant methods.

CCS Concepts

• **Hardware → Safety critical systems**; **Transient errors and upsets**.

Keywords

Accelerator, Isolation, SRAM-based FPGA

ACM Reference Format:
Sarah Azimi, Eleonora Vacca, Corrado De Sio, Luca Sterpone, and Andrea Portaluri. 2025. Mitigating Cross-Domain SEU Corruption in FPGA-Based AI Accelerators for Space Applications: Invited Paper. In *22nd ACM International Conference on Computing Frontiers (CF Companion '25), May 28–30, 2025, Cagliari, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3706594.3727947

CF Companion '25, Cagliari, Italy

ACM ISBN 979-8-4007-1393-4/25/05
https://doi.org/10.1145/3706594.3727947

1 Introduction

In recent years, the ability to reprogram their functionality has positioned Static Random Access Memory (SRAM)-based Field Programmable Gate Arrays (FPGAs) as key components in mission-critical systems, particularly where adaptability and extended operational lifetime are essential [15]. These devices support on-field reconfiguration, enabling dynamic changes to their logic to accommodate evolving tasks or altered environmental conditions. This flexibility is especially beneficial in aerospace, avionics, and increasingly in the automotive sector [19][4], where the integration of Artificial Intelligence (AI) and Machine Learning (ML) demands compact, reliable, and energy-efficient computing platforms.

FPGAs meet these demands by supporting highly parallel execution with low power consumption, outperforming traditional processors for many AI tasks [5][13]. This computing paradigm is central to modern AI hardware, including AMD's FPGA-accelerated platforms and Xilinx's AI Engines, which combine programmable logic with dedicated processing elements [25][24].

However, the growing deployment of FPGAs in safety- and mission-critical environments raises concerns about reliability in the presence of ionizing radiation. In such contexts, Single Event Effects (SEEs) are a major threat, leading to logic errors or functional interruptions [11]. SEEs are caused by the interaction between high-energy particles and semiconductor materials, producing faults such as Single-Event Transients, Latch-ups, Functional Interruptions, and most commonly, Single-Event Upsets (SEUs) [23][17]. SEUs occur when a transient pulse flips configuration bits in SRAM, potentially altering the behavior of logic elements or disrupting data flow.

Mitigation techniques like Triple Modular Redundancy (TMR) are widely adopted but may be insufficient in dense or multi-core designs where cross-domain errors occur—i.e., simultaneous faults in multiple redundant or parallel modules [14]. These scenarios can invalidate redundancy mechanisms and jeopardize computation, particularly in multi-core architectures designed for parallelism and performance [8][3].

In this paper, we propose and assess a set of layout strategies for multi-core FPGA designs aimed at increasing resilience against cross-domain SEUs. Our method leverages spatial resource isolation with different granularities to reduce the risk of shared routing

or logic corruption across cores[20][7].
To validate the approach, the layouts were implemented on a 16nm FinFET AMD UltraScale+ FPGA, and tested under proton irradiation (40–150 MeV) at the Paul Scherrer Institute (PSI) in Villigen, Switzerland. Experimental results show that fine-grained isolation can achieve up to a 36% reduction in the error cross-section compared to baseline placement strategies. Furthermore, the proposed strategies are compatible with traditional redundancy approaches, offering a path toward even greater robustness.

2 Related Works

A variety of techniques have been proposed to detect and mitigate Single Event Upsets (SEUs) at both software and hardware levels. Software-based methods include code replication, checkpoint-recovery systems, or combinations of both [23][22]. Hardware-level solutions typically involve module replication with majority voting (e.g., Triple Modular Redundancy) or scrubbing techniques, such as partial reconfiguration, to overwrite corrupted configuration memory (CRAM) when faults are detected [12].
Among known challenges, cross-domain errors have emerged as particularly critical. As highlighted in [14], such faults can affect multiple redundant modules simultaneously, undermining the effectiveness of standard fault-tolerant schemes. Several works [16][1] have investigated common routing resource sensitivity and developed custom tools to address errors spanning multiple domains. While promising, these approaches are often constrained to specific architectures or device families. Placement-aware mitigation strategies have recently gained traction. For example, the Xilinx Isolation Design Flow (IDF) enables designers to define "trusted routes" and isolate functional blocks to improve fault tolerance [26]. These techniques can complement other mitigation methods with minimal circuit-level changes. However, their applicability is limited—IDF demands detailed knowledge of proprietary FPGA internals and is tied to the Xilinx toolchain, making it infeasible for broader use. Efforts like [1] attempt to simplify the IDF for complex circuits, but still rely on Xilinx-specific features.
To date, no general placement methodology has been proposed for other FPGA vendors due to the lack of public CRAM mapping information and the absence of reliability-oriented strategies in mainstream commercial CAD tools. This highlights the need for more accessible, device-agnostic approaches to placement-level fault mitigation.

3 The proposed Methodology

To reduce cross-domain errors in multi-core FPGA designs, we developed layout strategies based on module isolation. This approach aims to prevent shared configuration and routing resources between computational blocks, thereby eliminating Single Points of Failure (SPoFs) that can compromise multiple modules simultaneously. Prior works indicate that keeping a spacing of 2 to 6 CLBs between blocks helps achieve effective isolation [26][1][10]. Although full physical isolation could theoretically prevent any shared bits, it often results in higher resource usage and increased latency, making it impractical for many real-world applications.

In this context, we propose two isolation strategies—fine-grained and coarse-grained—to explore different trade-offs between placement strictness and implementation overhead. By increasing the physical separation between cores, these layouts lower the chance of shared logic or routing and thus improve resilience to cross-domain faults. The effectiveness of each approach is evaluated based on the balance between isolation granularity and system efficiency.

3.1 Fault Model

The configuration memory (CRAM) of SRAM-based FPGAs is a volatile storage area that holds the data defining the implemented design. This configuration is loaded via a bitstream—a binary file containing directives for each programmable resource, such as Configurable Logic Blocks (CLBs), Block RAMs (BRAMs), and routing elements. The bitstream encodes settings like LUT truth tables, BRAM contents, and interconnect paths through Programmable Interconnect Points (PIPs). A single bit-flip in these routing directives can lead to critical faults such as open circuits, disrupting the data flow as illustrated in Fig. 1.

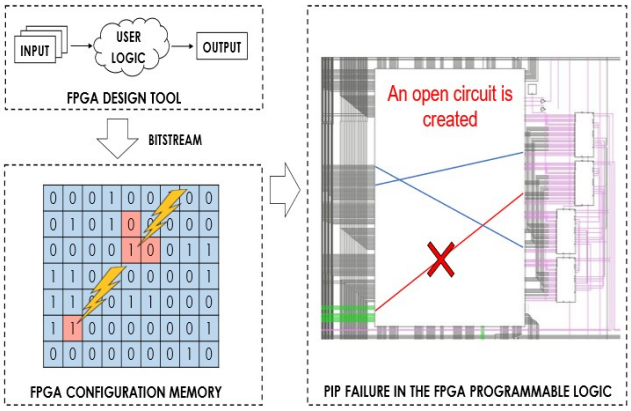

Figure 1: SEUs corrupting the circuit implemented on the FPGA by bit-flipping the content of the configuration memory, causing interconnection points to become open circuits and so compromising the correctness of the application.

A widely used mitigation strategy for radiation-induced faults is hardware redundancy, including techniques like Duplication with Comparison (DWC) and Triple Modular Redundancy (TMR). These rely on replicating computation across multiple domains and using voting logic to select the correct output, even if one domain fails. However, this approach is vulnerable to cross-domain errors, where a single SEU affects a resource or bit shared between multiple domains—such as a common routing path—thereby invalidating the redundancy and compromising the system's reliability.

3.2 Isolation Granularity

FPGA designs often rely on vendor-provided Intellectual Property (IP) blocks, which are pre-packaged and lack internal visibility—essentially acting as black boxes. This limits the feasibility of isolating individual logic resources within them. To address this, we extended the concept of isolation to the core level, treating each computational unit as an indivisible block. In this context, fine-grained isolation refers to keeping each core physically separate, while coarse-grained isolation allows controlled resource sharing among groups of cores.

Modern FPGA CAD tools offer flexible placement and routing control, enabling designers to specify exact coordinates for logic and interconnects. Using placement blocks (pblocks), one can constrain modules to defined regions of the FPGA fabric. As illustrated in Fig. 2, these isolation strategies differ in routing density and overhead: default layouts prioritize compactness but increase shared interconnects; fine-grained layouts eliminate shared routes at the cost of higher area usage; and coarse-grained layouts offer a practical compromise between fault tolerance and efficiency.

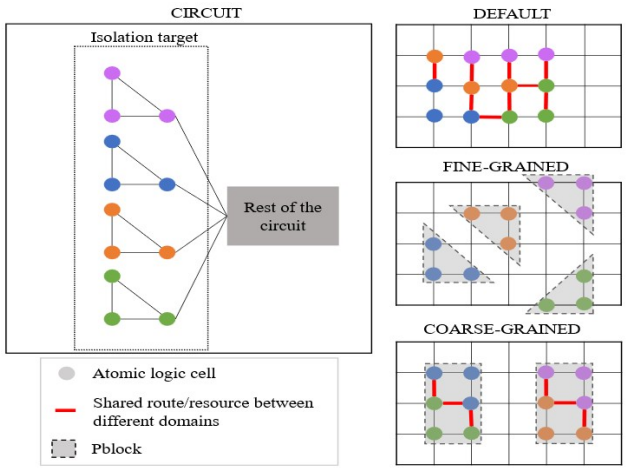

Figure 2: Schematic view of the developed application benchmark.

3.3 Application Benchmark

To emulate the behavior of AI-oriented multi-core accelerators, we designed a custom benchmark capable of managing both data movement and computation. The system integrates programmable logic with a hardwired microprocessor, reflecting typical heterogeneous FPGA-based computing platforms. In the programmable logic, we implemented 32 CORDIC (COordinate Rotation DIgital Computer) cores [18], widely used for hardware-efficient computation of transcendental functions, square roots, and vector rotations using only additions, subtractions, and bit-shifts.

Data exchange between memory and cores is handled by two DMA engines, configured in full 16-channel Scatter-Gather mode with maximum burst length, enabling parallel communication with all 32 CORDIC instances. Each core's output is tagged with a channel ID for identification and error tracking. The microprocessor interfaces with the DMA modules through AXI High-Performance ports. Buffer Descriptors (BDs) for memory transfers—both MM2S and S2MM—are allocated in dedicated BRAMs, ensuring low-latency access and efficient scrubbing. DMA interrupt signals are aggregated and monitored to identify faults or anomalies in data movement.

A lightweight bare-metal routine runs on the processor to manage BDs, handle DMA interrupts, and periodically refresh memory to mitigate potential corruption. It also performs software-based Triple Modular Redundancy (TMR) voting on core outputs, enabling the identification and logging of faulty cores during radiation testing. The overall architecture of this benchmark is depicted in Fig. 3.

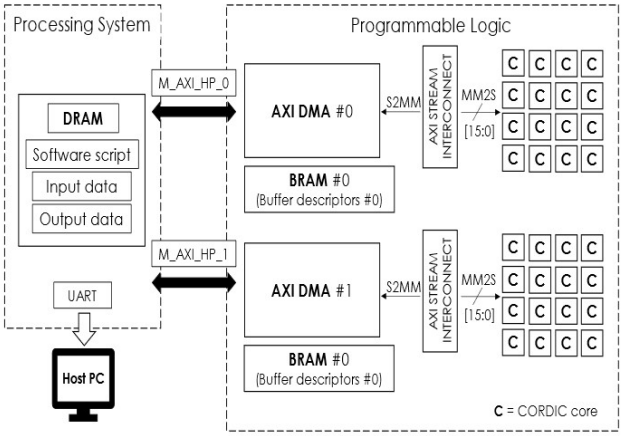

Figure 3: Schematic view of the developed application benchmark.

3.4 Applied Solution

Using the developed hardware benchmark, we implemented three layout configurations to evaluate the effectiveness of the proposed isolation strategies from previous section. Two of these leverage isolation with different granularities—Matrix (fine-grained) and Columns (coarse-grained)—while Default follows the standard placement approach and serves as a baseline for comparison.

- In the Default configuration, the FPGA CAD tool performs automatic placement and routing without constraints. As a result, cores are freely distributed across the fabric, potentially sharing logic and routing resources, and offering no protection against cross-domain errors.
- The Matrix layout applies strict fine-grained isolation: all 32 cores are placed in individual, non-overlapping pblocks arranged in an 8×4 grid. Each pblock measures 11×11 CLBs and is separated from its neighbors by a 2-CLB gap in both directions. This layout ensures that no routing or configuration bits are shared between cores. Core placement follows a row-major order (e.g., row 0: cores 0–3, row 1: cores 4–7, etc.), allowing a consistent spatial mapping for analysis.

- The Columns layout adopts a coarse-grained approach, grouping the 32 cores into 8 vertical pblocks, each containing 4 cores. While cores within the same column may share some resources, each column remains isolated from the others by a 2-CLB gap. Each pblock spans 44×11 CLBs—four times the size of a Matrix pblock—to accommodate the grouped cores. Core IDs are assigned in a column-major order (e.g., column 0: cores 0–3, column 1: cores 4–7, and so on).

To ensure a fair comparison, both Matrix and Columns layouts were designed using the smallest possible pblocks that still contain all required resources. This guarantees that the total number of CLBs used across configurations remains consistent, minimizing design bias due to area differences. Fig. 4.
illustrates the implemented layout views in Xilinx Vivado for all three configurations.

4 Experimental Setup

To assess the resilience of the proposed solutions against radiation-induced cross-domain errors, we conducted a dedicated radiation test campaign. Such tests closely replicate space-like conditions by exposing devices to accelerated high-energy particle beams, which emulate the effects of ionizing radiation. These facilities allow precise control over particle energy and fluence, directing the beam onto specific targets such as boards, chips, or subcomponents.
Due to the high-radiation environment, data acquisition and power control were fully automated using a custom experiment platform developed in-house. The following subsections describe the experimental setup and testing infrastructure in detail.

4.1 Radiation Test Setup and Device Under Test

The board was mounted on an adjustable frame and aligned with the beam without removing any components, preserving its Commercial Off-The-Shelf (COTS) state to reflect real-world use. Power and data lines connected to a control room enabled reconfiguration, output logging, and power cycling. A Python-based experiment manager, built on the PyXEL framework [6], handled automated data collection and CRAM readback every 10 seconds. In case of UART failure, the system reset the board and reloaded both the bitstream and processor state. Beam parameters were tuned to induce 2–5 bit-flips per readback, simulating realistic LEO radiation conditions [21].
The Device Under Test (DUT) was the ZCU104 Evaluation Kit, featuring a Xilinx Zynq UltraScale+ MPSoC built on 16nm FinFET technology. It includes a quad-core ARM Cortex-A53 processor—used to run the software routine—a dual-core Cortex-R5, and a Mali-400 GPU.
Proton beam parameters used in the campaign are summarized in Table 1. Each layout was tested across four beam runs per energy level, with possible system failures monitored as defined in Section V.A. To account for flux variability, the table reports average flux values per energy. The higher fluence at 101.34 MeV reflects additional tuning time during initial beam setup.

Table 1: Energy levels, average fluxes, and total fluences measured at the PSI Facility.

E [MeV]	Flux [$cm^{-2}s^{-1}$]	Std. dev. [$cm^{-2}s^{-1}$]	Fluence [cm^{-2}]
43.20	$3.940 \cdot 10^{7}$	$4.163 \cdot 10^{4}$	$3.506 \cdot 10^{10}$
50.80	$4.034 \cdot 10^{7}$	$9.192 \cdot 10^{4}$	$4.723 \cdot 10^{10}$
69.71	$4.191 \cdot 10^{7}$	$1.400 \cdot 10^{5}$	$7.195 \cdot 10^{10}$
101.34	$3.986 \cdot 10^{7}$	$9.582 \cdot 10^{4}$	$2.782 \cdot 10^{11}$
151.18	$4.094 \cdot 10^{7}$	$4.950 \cdot 10^{3}$	$1.506 \cdot 10^{9}$

4.2 Experimental Results

Table I reports the resource utilization and number of Essential Bits for the three implemented layouts. Essential Bits, as identified by the Xilinx Vivado tool, represent a subset of configuration memory (CRAM) bits that are critical to circuit behavior—any corruption of these bits may lead to functional errors. The results show negligible differences across all layouts, indicating that applying spatial isolation to the cores does not introduce any significant overhead in terms of logic utilization or critical configuration bits.
It is worth noting that the placement constraints were applied exclusively to the computational cores. Supporting components such as DMA engines, BRAMs, and interconnect controllers were left unconstrained. However, care was taken to ensure that these modules were placed outside the vicinity of the isolated core regions, so as not to compromise the separation or introduce unintended resource sharing.
To assess the radiation sensitivity of the overall designs, we computed the SEU cross-section per CRAM bit for each layout. The cross-section quantifies the likelihood that a configuration bit will experience a fault under a given particle fluence and is calculated by dividing the number of observed SEUs by the total fluence per energy level. These values were normalized by the number of unmasked CRAM bits—i.e., those bits that are not hidden by the bitstream masking process and are considered sensitive to radiation. Masked bits typically configure dynamic elements like flip-flops, LUTRAMs, and BRAMs, and are excluded to avoid counting changes that may occur due to normal operation rather than radiation effects.

Fig. 5 shows that all three layouts exhibit a consistent SEU cross-section of approximately 2.8×10^{-16} cm^2/bit, which is in line with values reported in prior experiments for the same 16 nm FinFET FPGA technology [9]. As expected, the cross-section is largely independent of the logical design or layout strategy, since it is a property of the underlying device architecture. Nevertheless, this consistency is valuable, as it confirms that each configuration was exposed to an equivalent radiation environment and SEU rate. This validates the fairness of the upcoming comparison between layouts, focusing on how isolation impacts the behavior and reliability of individual computational cores under radiation-induced stress.

5 Conclusion

This work introduced two implementation strategies for multi-core accelerators on FPGAs: fine-grained and coarse-grained isolation. These approaches isolate resources at different granularities to evaluate their resilience to radiation-induced SEUs. A proton irradiation

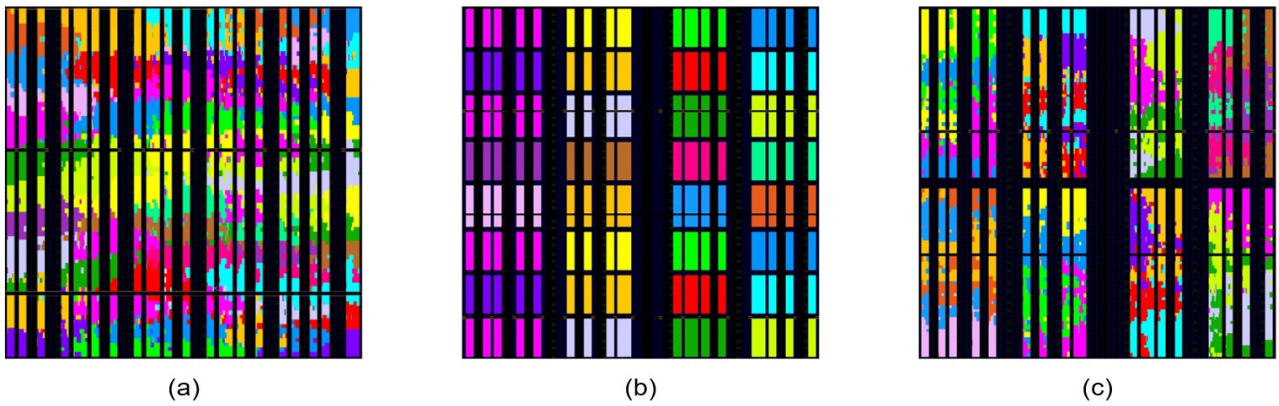

Figure 4: Vivado implementation views* of the three layouts: (a) Default, (b) Matrix, and (c) Columns. A different color is assigned to each core.**

*** The view does not exactly reflect the physical implementation on the device.**

**** The colors may be repeated due to the limited palette available in Vivado.**

Table 2: Resources utilization and essential bits for Default, Matrix, and Columns layouts on ZCU104. The values include used and available resources, and their utilization percentages.
All data were extracted using the Xilinx Vivado tool after implementation.

Layout	LUT			FF			BRAM			Essential Bits		
	Used [#]	Avail. [#]	Utiliz. [%]	Used [#]	Avail. [#]	Utiliz. [%]	Used [#]	Avail. [#]	Utiliz. [%]	Used [#]	Avail. [#]	Utiliz. [%]
Default	30,633	230,400	13.30	34,032	460,800	7.39	14	312	4.39	10,791,647	131,763,200	8.19
Matrix	30,631	230,400	13.29	34,031	460,800	7.39	14	312	4.39	11,000,622	131,763,200	8.35
Columns	30,638	230,400	13.29	34,032	460,800	7.39	14	312	4.39	11,142,122	131,763,200	8.46

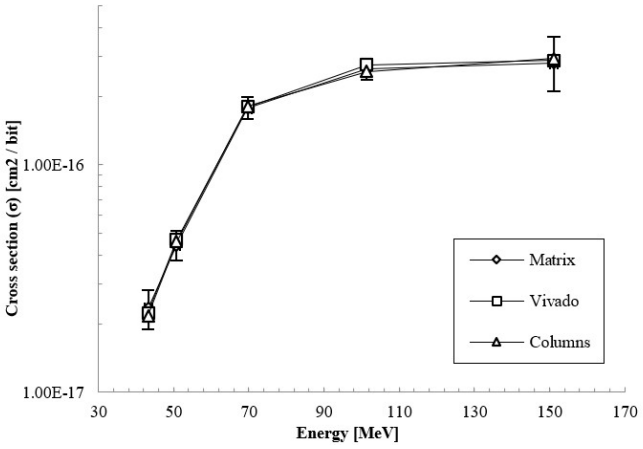

Figure 5: SEU cross section per unmasked CRAM bit for the three layouts (95% confidence error bars [2]).

campaign (40–150 MeV) was conducted to assess both the overhead and fault tolerance of each layout. Results showed that resource isolation can reduce shared routing and dependency on neighboring logic by up to 36%. While not designed to prevent bitflips directly, the method is compatible with existing fault-tolerant techniques like DWC or TMR, offering a complementary reliability boost. Future work includes developing a tool to automate these placement strategies.

Acknowledgments

The authors want to thank Dr. Wojciech Hajdas and his team at the PSI Proton Irradiation Facility in Villigen, Switzerland, for their precious time and support during the radiation experiment.

References

[1] S. Azimi A. Portaluri, C. De Sio and L. Sterpone. 2021. A New Domains-based Isolation Design Flow for Reconfigurable SoCs. In *IEEE 27th International Symposium on On-Line Testing and Robust System Design (IOLTS).* doi:10.1109/IOLTS52814.2021.9486687

[2] European Space Agency. 2014. Single Event Upset Characterization of the Zynq UltraScale+ MPSoC Using Proton Irradiation. In *Single Event Effects Test Method and Guidelines.*

[3] [8] S. Azimi and L. Sterpone. 2020. Digital Design Techniques for Dependable High Performance Computing. In *IEEE International Test Conference (ITC).* doi:10.1109/ITC44778.2020.9325281

[4] L. Bozzoli B. Du L. Sterpone C. De Sio, S. Azimi. 2019. Radiation-induced Single Event Transient effects during the reconfiguration process of SRAM-based FPGAs. In *Microelectronics Reliability.* doi:10.1016/j.microrel.2019.06.034

[5] L. Sterpone C. De Sio, S. Azimi. 2020. An Emulation Platform for Evaluating the Reliability of Deep Neural Networks. In *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems.* doi:10.1109/DFT50435.2020.9250872

[6] L. Sterpone D. M. Codinachs C. De Sio, S. Azimi and F. Decuzzi. 2023. PyXEL: Exploring Bitstream Analysis to Assess and Enhance the Robustness of Designs on FPGAs. In *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD).* doi:10.1109/SMACD58065.2023.10192116

[7] S. Azimi C. De Sio and L. Sterpone. 2020. An Emulation Platform for Evaluating the Reliability of Deep Neural Networks. In *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*. doi:10.1109/DFT50435.2020.9250872

[8] S. Azimi C. De Sio and L. Sterpone. 2022. FireNN: Neural Networks Reliability Evaluation on Hybrid Platforms. *IEEE Transactions on Emerging Topics in Computing* (2022). doi:10.1109/TETC.2022.3152668

[9] V. Kirischian D. M. Hiemstra and J. Brelski. 2017. Single Event Upset Characterization of the Zynq UltraScale+ MPSoC Using Proton Irradiation.

[10] S. Azimi E. Vacca, C. De Sio. 2022. Layout-oriented Radiation Effects Mitigation in RISC-V Soft Processor. In *ACM International Conference on Computing Frontiers*. doi:10.1145/3528416.3530984

[11] C. Cai et al. 2019. SEE Sensitivity Evaluation for Commercial 16 nm SRAM-FPGA. *Electronics* 8 (2019). doi:10.3390/electronics8121531

[12] L. A. Cardona et al. 2017. A novel tool-flow for zero-overhead cross-domain error resilient partially reconfigurable X-TMR for SRAM-based FPGAs. *Journal of System Architecture* (2017). doi:10.1016/j.sysarc.2017.10.009

[13] A. Zaki F. Taher and H. Elsimary. 2016. Design of low power FPGA architecture of image unit for space applications. In *IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*. doi:10.1109/MWSCAS.2016.7870001

[14] P. Graham J. Krone M. Caffrey H. Quinn, K. Morgan and K. Lundgreen. 2007. Domain Crossing Errors: Limitations on Single Device Triple-Modular Redundancy Circuits in Xilinx FPGAs. *IEEE Transactions on Nuclear Science* (2007). doi:10.1109/TNS.2007.910870

[15] B. Dum S. Azimi L. Sterpone. 2015. Radiation-induced single event transients modeling and testing on nanometric flash-based technologies. In *Microelectronics Reliability*. doi:10.1016/j.microrel.2015.07.035

[16] A. Keller M. Cannon and M. Wirthlin. 2018. Improving the Effectiveness of TMR Designs on FPGAs with SEU-Aware Incremental Placement. In *IEEE Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. doi:10.1109/FCCM.2018.00031

[17] L. Sterpone N. Battezzati and M. Violante. [n. d.]. Reconfiguable Field Programmable Arrays for Mission-Critical Applications. In *Springer New York*. doi:10.1007/978-1-4419-7595-9

[18] K. Raj R. Tomar, P. K. Singh. 2013. A Review of CORDIC Algorithms and Architectures with Applications for Efficient Designing. *International Journal of Scientific and Engineering Research* (2013). doi:10.1109/TNS.2018.2877579

[19] Romoth and M. Porrmann. 2017. Survey of FPGA applications in the period 2000 – 2015. *Technical Report* (2017).

[20] B. Du L. Boragno S. Azimi, L. Sterpone. 2018. On the analysis of radiation-induced Single Event Transients on SRAM-based FPGAs. In *Microelectronics Reliability*. doi:10.1016/j.microrel.2018.07.135

[21] E. G. Stassinopoulos and J. P. Raymond. 2023. The space radiation environment for electronics. In *IEEE proceeding)*. doi:10.1109/5.90113

[22] L. Sterpone and M. Violante. 2005. A new analytical approach to estimate the effects of SEUs in TMR architectures implemented through SRAM-based FPGAs. *IEEE Transactions on Nuclear Science* (2005). doi:10.1109/TNS.2005.860745

[23] M. Wirthlin. 2015. High-Reliability FPGA-Based Systems: Space, High-Energy Physics, and Beyond. In *Proceedings of the IEEE*. doi:10.1109/JPROC.2015.2404212

[24] Xilinx. 2020. Xilinx AI Engines and Their Applications.

[25] Xilinx. 2021. DPUCZDX8G for Zynq UltraScale+ MPSoCs.

[26] Xilinx. 2021. Isolation Design Flow for UltraScale+ FPGAs and Zynq UltraScale+ MPSoCs. In *Application Note XAPP1335*.