Self-Contained Configuration Scrubbing in Xilinx FPGAs for On-detector Applications

R. Giordano[1], S. Perrella[1], D. Barbieri[1]

[1] Università di Napoli "Federico II", I-80126 Napoli, Italy and
INFN Sezione di Napoli, I-80126, Napoli, Italy

Summary

Off-detector electronics in TDAQ systems of HEP experiments is often implemented by means of SRAM-based FPGAs, due their high-performance in real-time data processing and transfer. On-detector usage of FPGAs is mostly limited by their sensitivity to radiation-induced upsets in the configuration (Fig. 1). In fact, these effects may disrupt the intended functionality and there is the need to correct, i.e. scrub, the configuration memory periodically.

In this work, we describe a simple-yet-effective scrubber core designed for Xilinx FPGAs based on configuration redundancy. The core does not require any external memory for hosting the golden configuration, yet the redundancy make it possible to correct a significant number of upsets per frame. The scrubber is able to correct the configuration of a given design and its own.

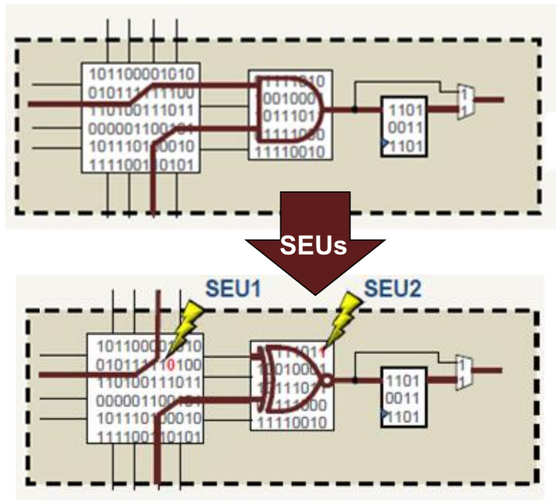
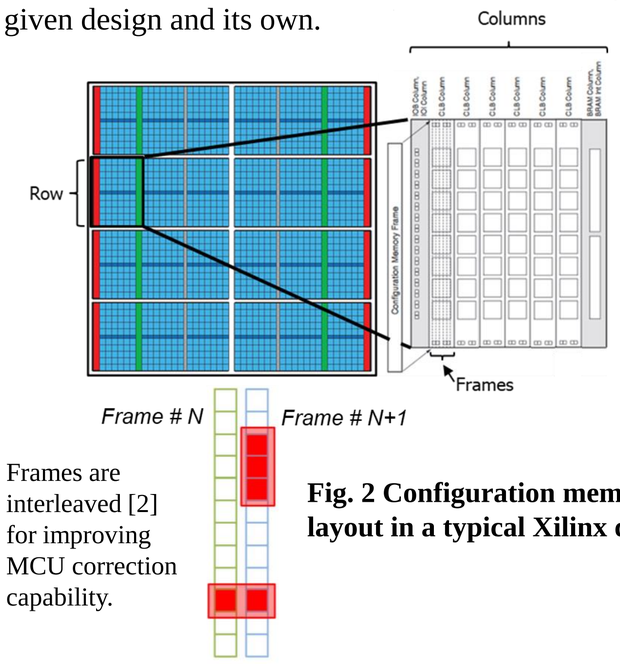


Fig. 1 Impact of configuration SEUs on FPGA functionality.



Frames are interleaved [2] for improving MCU correction capability.

Fig. 2 Configuration memory layout in a typical Xilinx device.

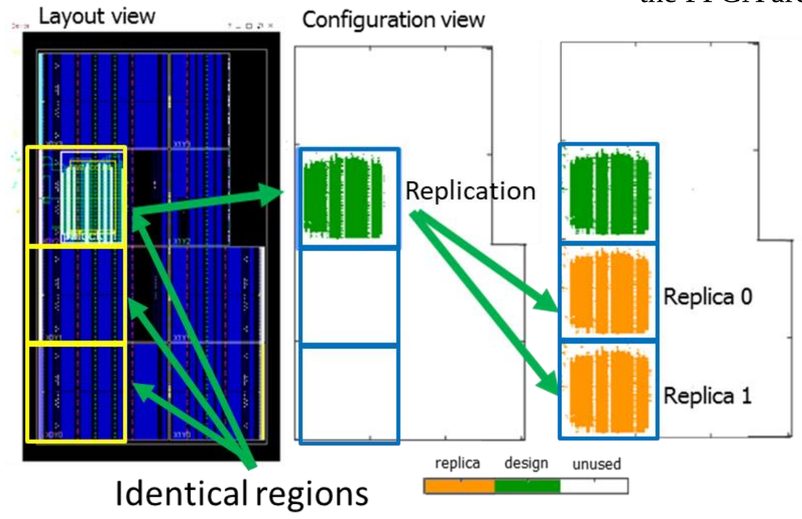


Fig. 3 Generation of redundant configuration.

1. Generation of a Redundant Configuration

Modern FPGAs enable read/write access to configuration frames (Fig. 2) via JTAG or other IO ports. We configure the FPGA with the design bitstream and then, by means of the JTAG port, we generate redundant copies into the frames of unused identical regions (Fig. 3) [1]. This makes it possible to compare and correct copies during the FPGA operation.

Redundant copies do not receive clock, there is no increase in dynamic power consumption, only a small increase in quiescent power consumption (Fig. 4). This approach only requires a configuration access port, and therefore it supports multiple Xilinx FPGA families. The geometric separation between redundant bits enhances the correctability of MCUs. This approach is unrelated to Triple Modular Redundancy. At the end of Configuration Redundancy Generation, a list of redundant frame sequences and a list of frames sequences pertaining to unused regions in the FPGA are produced.

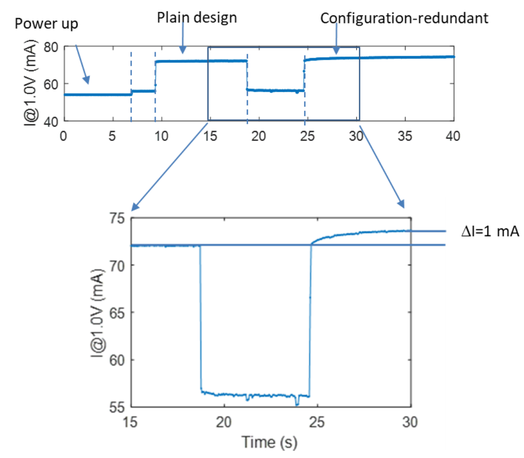
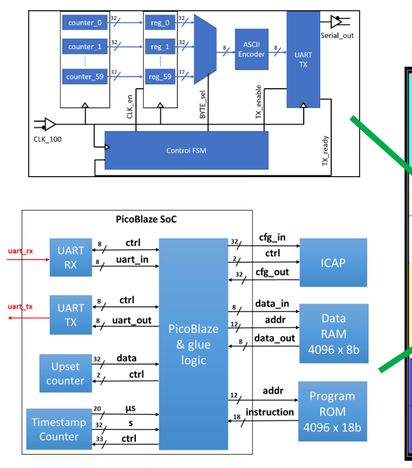


Fig. 4 Power consumption at the VCCINT power domain during redundancy generation.

2. Scrubber Architecture & Implementation

- Test firmware w/ benchmark design and scrubber (Fig. 5)
- Fully redundant configuration (Fig. 6)
- Scrubber w/ tiny logic footprint (nearly 500 slices in a 7-Series FPGA)
- Based on Xilinx PicoBlaze & ICAP
- Supports correction of the full device
- No custom design flow needed
- Portable on most Xilinx devices
- Provides detailed information about detected upsets (FRAD, offset, timestamp, polarity)



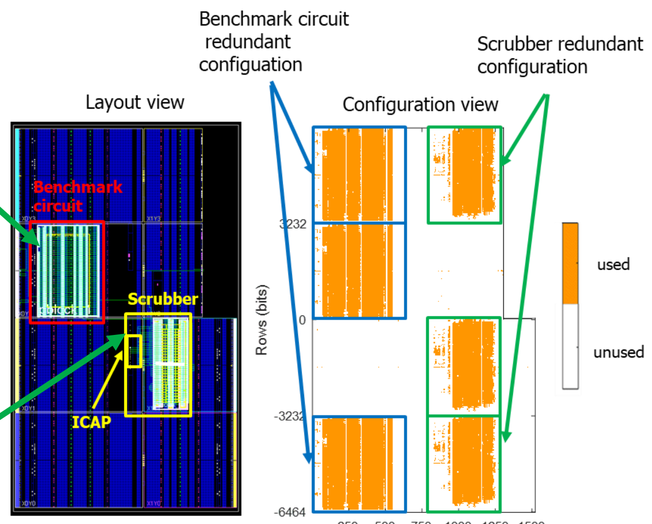


Fig. 5 Scrubber and benchmark circuit block diagrams and implementation layout.

Fig. 6 Configuration of the test firmware. Each pixel is a cluster of 32 bits by 5 frames.

3. Proton Beam Test

- Test with a 62-MeV proton beam at Superconducting cyclotron at Laboratori Nazionali del Sud – Catania, Italy (Fig. 7)
- Total fluence $6.7 \cdot 10^{10}$ cm^{-2}, 110 runs, flux between $2.2 \cdot 10^{7}$ cm^{-2} s^{-1} and $3.5 \cdot 10^{7}$ cm^{-2} s^{-1}

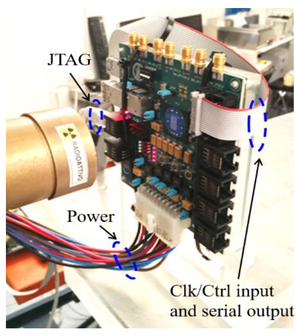


Fig. 7 DUT board facing the beam pipe at LNS.

- Custom DUT board FPGA loaded with test firmware (Fig. 8)
- Tester board (Xilinx KC705) runs same benchmark circuit as DUT and it verifies DUT output
- Multichannel Power Analyzer (PA) powers DUT board
- Test controller logs upsets from scrubber, functionality test results from tester board, and power logs from PA
- Remote PC accesses test controller from control room

4. Test Results

The scrubber has been proven to

- Repair the configuration (any number of upsets per frame) (Fig. 9 and 10)
- Keep the power consumption stable within few mA at any power input of the FPGA
- Improve the mean absorbed proton fluence before a persistent failure of a benchmark design by 42% (benchmark w/ TMR) and 290% (benchmark w/out TMR) (Fig. 11)

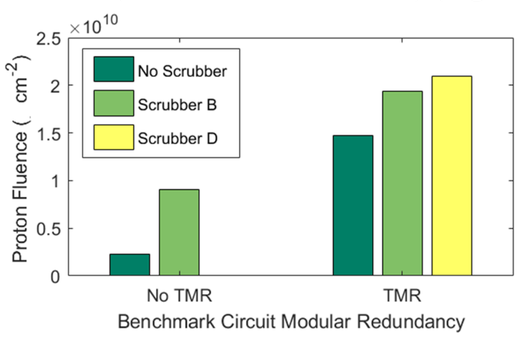


Fig. 11 Proton fluence tolerated by the firmware before failure.

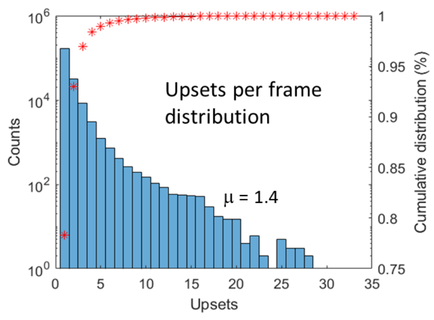


Fig. 9 Histogram of the number of detected and corrected upsets per frame.

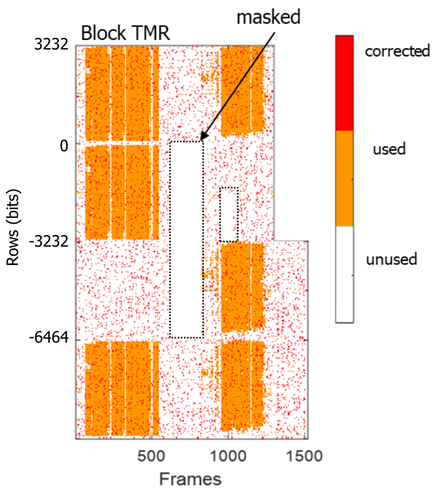


Fig. 10 Configuration bitmap w/ detected and corrected upsets. Each pixel is a cluster of 32 bits by 5 frames.

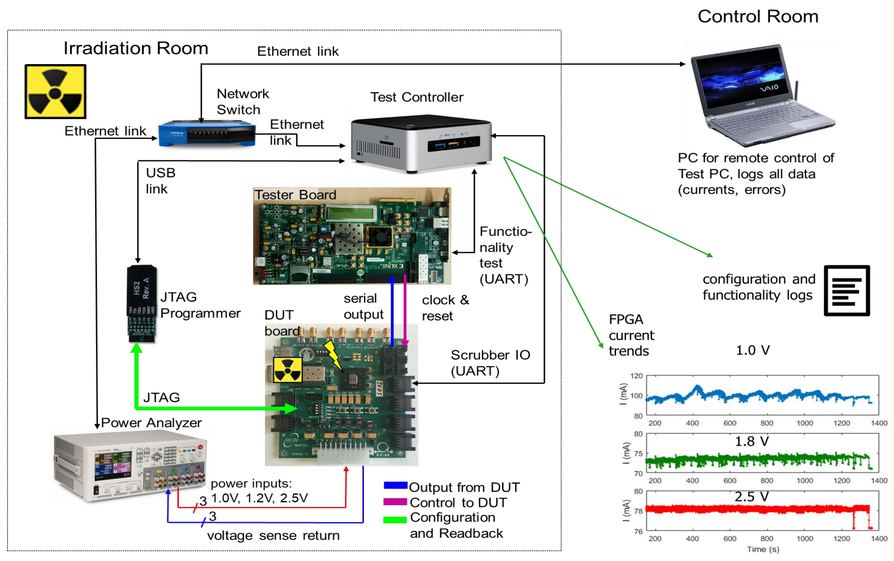


Fig. 8 Test Setup.

References

[1] R. Giordano et al., IEEE Trans. on Nucl. Sci., vol. 64, no. 9, pp. 2497-2504, Sept. 2017
[2] M. J. Wirthlin et al., 2014 JINST 9 C01025

This work is part of the ROAL project (grant no. RBSI14JOUV) funded by the Scientific Independence of Young Researchers (SIR) 2014 program of the Italian Ministry of Education, University and Research (MIUR).

www.roalproject.it