# Design of Finite State Machines for SRAM-based FPGAs operated in radiation field


**M. Lupi**[a,b], **P. Giubilato**[c], **M. Bonora**[d], **K. M. Sielewicz**[e]

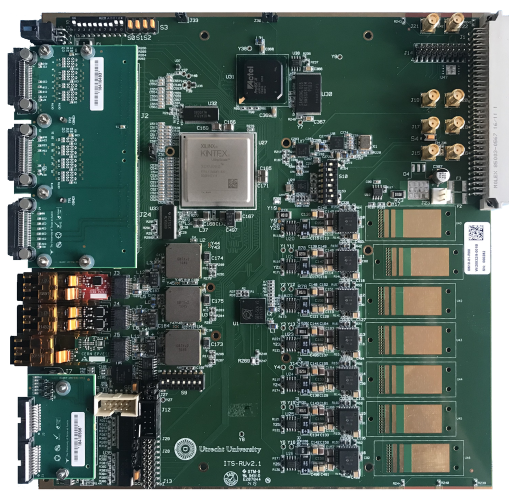
[a]CERN, Geneva (CH), [b]Goethe University, Frankfurt am Main (DE), [c]Università e INFN, Padova (IT), [d]Salzburg University, Salzburg (AT), [e]previously CERN, Geneva (CH), [e]previously Warsaw University of Technology, Warsaw (PL)

matteo.lupi@cern.ch


## Introduction

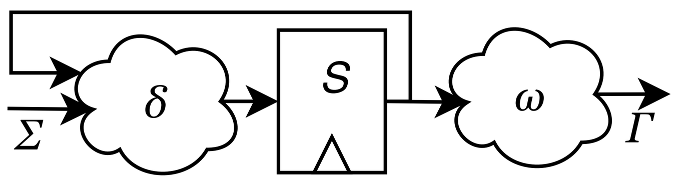
During the current CERN Large Hadron Collider shutdown period, the ALICE experiment is upgrading its Inner Tracking System (ITS) [1]. The new ITS uses 24120 ALPIDE chips: custom large area CMOS Monolithic Active Pixel Sensors (MAPS) [2]. The readout system, composed of **192** identical **Readout Units (RU)**, has **complete control over all sensor operations**, including power management. Its reliability is, therefore, critical for the correct operation of the entire ITS. The RUs will be placed at about five meters from the interaction point, along the beam axis, and at a radial distance of about one meter. The expected Total Ionising Dose (TID) for the entire detector life cycle is about 10 krad(Si), including a safety factor of ten, which does not raise concerns since all the system components have been verified against this dose.

Conversely, **the expected flux of High Energy Hadrons** with sufficient energy ($>$ 20 MeV) to **induce Single-Event Effects (SEEs)** in microelectronic devices is of the order of $\mathbf{1 \cdot 10^3\,cm^{-2}\,s^{-1}}$, posing a challenge to the utilisation of commercial, SRAM-based FPGAs. Irradiation tests showed that the whole system of 192 RUs, each employing a Xilinx Kintex UltraScale XCKU060 FPGA, **will experience**, on average, **an SEE affecting the FPGA configuration RAM (CRAM) every eight seconds**. An external scrubbing sub-system, driven by a flash-based FPGA, ensures the long-term stability of the FPGA design. However, the SRAM-based FPGA design needs to deal with errors until the scrubbing corrects them. Previous publications [3] showed how to efficiently protect combinatorial and sequential networks without feedback. **Finite State Machines (FSMs)** represent a **fundamental building block** of an FPGA design: since they need to retain a state, it is mandatory that they can continue operating between scrub cycles.

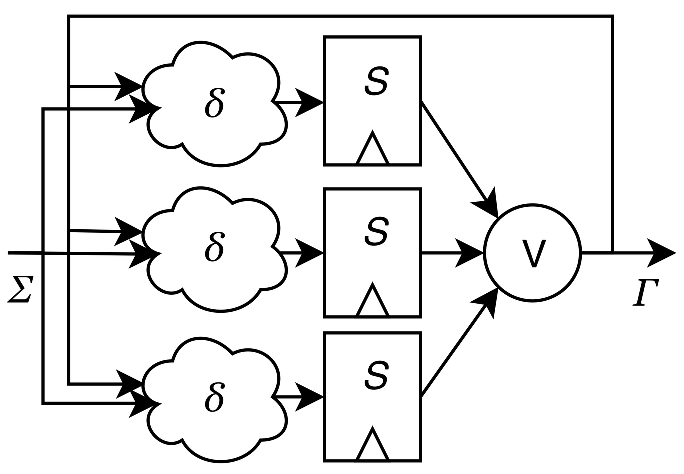


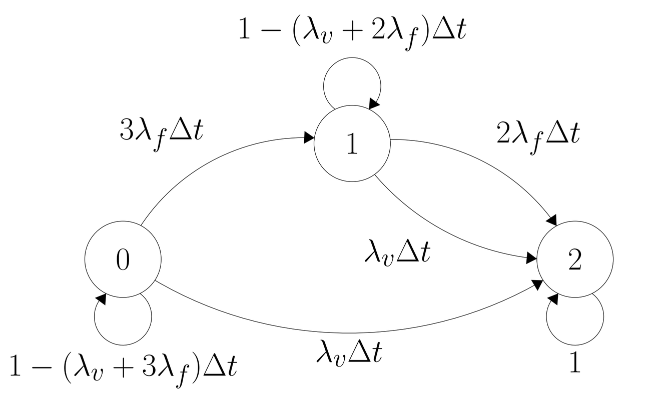
ITS Readout Unit

## Theoretical approach




The **reliability** of a state machine can be **modelled** using **time-continuous Markov chains** [4]. The model proposed, first studies a single bit FSM, protected using different mitigation techniques: **no mitigation**, **Triple Modular Redundancy** (TMR) with single and triplicated voter, and **Hamming** with Single Error Correction (SEC). It takes into account different failure rates for logic $\lambda_f$ and voters $\lambda_v$, and it can also account for scrubbing correction rate $\mu$ if the state of the FSM is not lost due to the upset.



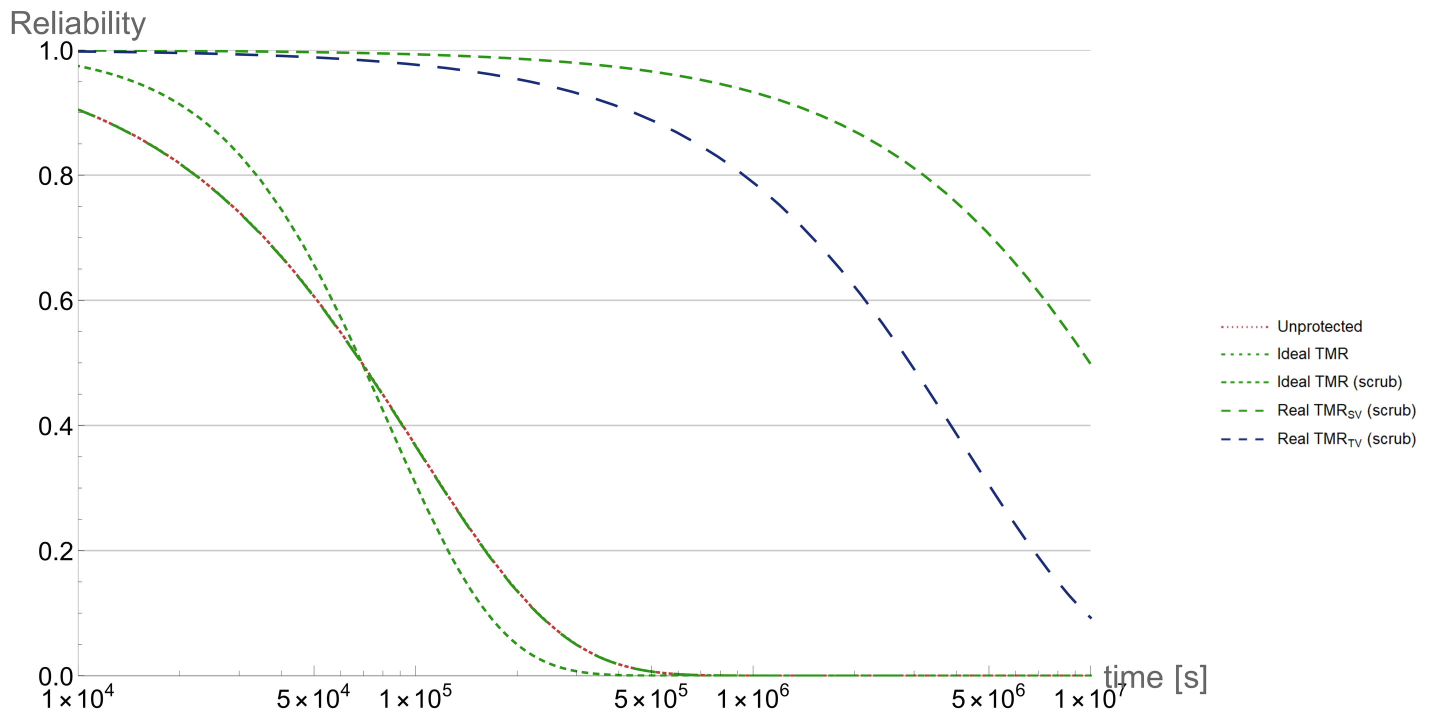

In the base example provided, a **triplicated FSM protected with a single voter** with non-negligible cross-section is shown. The associated Markov chain can be drawn assuming three different states: one where the FSM is working, one where one of the three TMR branches is broken, and one where the FSM is not working any longer. The latter happens either when the voter is broken or when two branches have an upset. From the Markov chain a state transition matrix, $\underline{\underline{T}}$ can be derived, indicating the system evolution as a function of time. Solving the differential-equation system, the probability of the system being in a determinate state can be found. The reliability of the system, $R(t)$ is the probability of the system being in the state $0$ or $1$. As it can be observed in (2) **the reliability is the product of the reliabilities of the sub-system**: this is the expected behaviour of a series of sub-systems. The first term is the reliability of the voter, the second term the reliability of a triplicated system.

$$\underline{\underline{T}}(\Delta t) = \begin{bmatrix} 1-3\lambda_f\Delta t-\lambda_v\Delta t & 3\lambda_f\Delta t & \lambda_v\Delta t \\ 0 & 1-2\lambda_f\Delta t-\lambda_v\Delta t & 2\lambda_f\Delta t+\lambda_v\Delta t \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$\mathbf{R(t) = e^{-\lambda_v t}(3e^{-2\lambda_f t} - 2e^{-3\lambda_f t})} \tag{2}$$

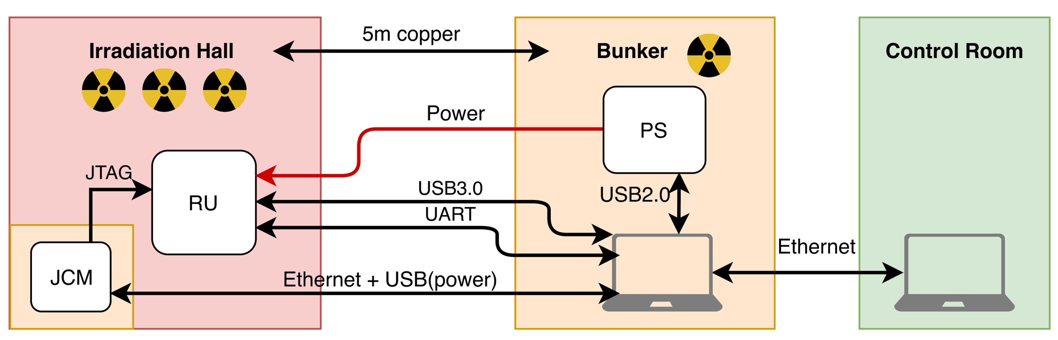
The reliability of multi-bit FSMs can be calculated as the reliability of the series (if one fails, the FSM fails) of single-bit FSMs. The reliability of the FSMs protected with Hamming, required a similar approach to the one described, but it requires studying the FSM knowing the number of state bits.

The plot (below) shows the reliability function of an eight-state FSM protected with different radiation effect mitigation techniques and logic upset rate of $\lambda_f = 1 \cdot 10^{-12}$. In particular, **the effect of the voter cross-section and the scrubbing are studied**. In the case of the unprotected FSM, scrubbing has no effect. The ideal TMR has a negligible voter cross-section, whereas, for the real TMR **the voter has the same cross-section as the logic**, which represents a realistic scenario in SRAM-based FPGAs. The ideal TMR shows no difference between single and triple voter implementation, whereas, the difference is significant when scrubbing is active.



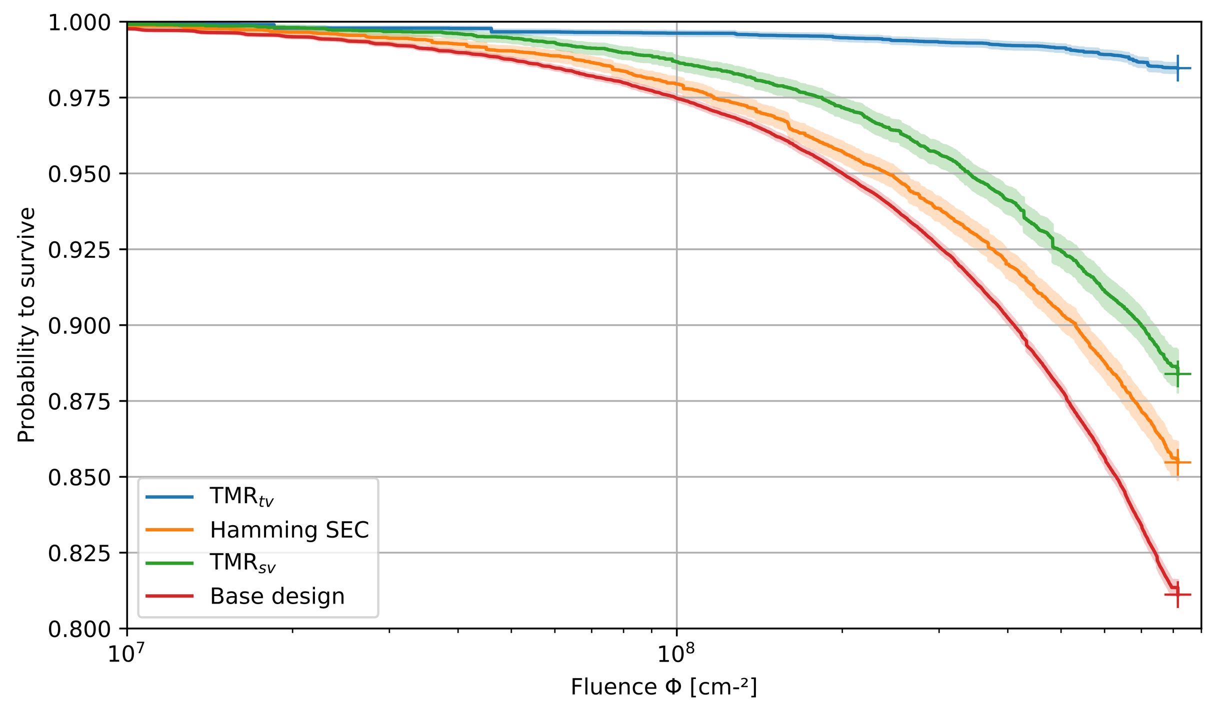

## Experimental approach

The different radiation effect mitigation techniques have been validated on a Xilinx Kintex-7 XC7K325T FPGA by **fault injection** and **proton irradiation**. The **fault injection** [3], was executed in the laboratory by toggling a CRAM bit during the FPGA operation and subsequently restoring its value. The number of errors corresponds to an equivalent fluence of $\Phi_{equivalent} = 8.2 \cdot 10^8\,\mathrm{cm}^{-2}$, *i.e.* **227 h of operation of a single RU**, or 1.18 h of the whole system. The **20 MeV proton irradiation** was run with a **flux of $\mathbf{1 \cdot 10^6\,cm^{-2}\,s^{-1}}$**, in order for the mean time between CRAM upset to match the scrubbing cycle. The test flux was 1000 time higher than the target application flux, but it was used to collect statistics in a reasonable time and to study how the design operates when CRAM upsets accumulate.

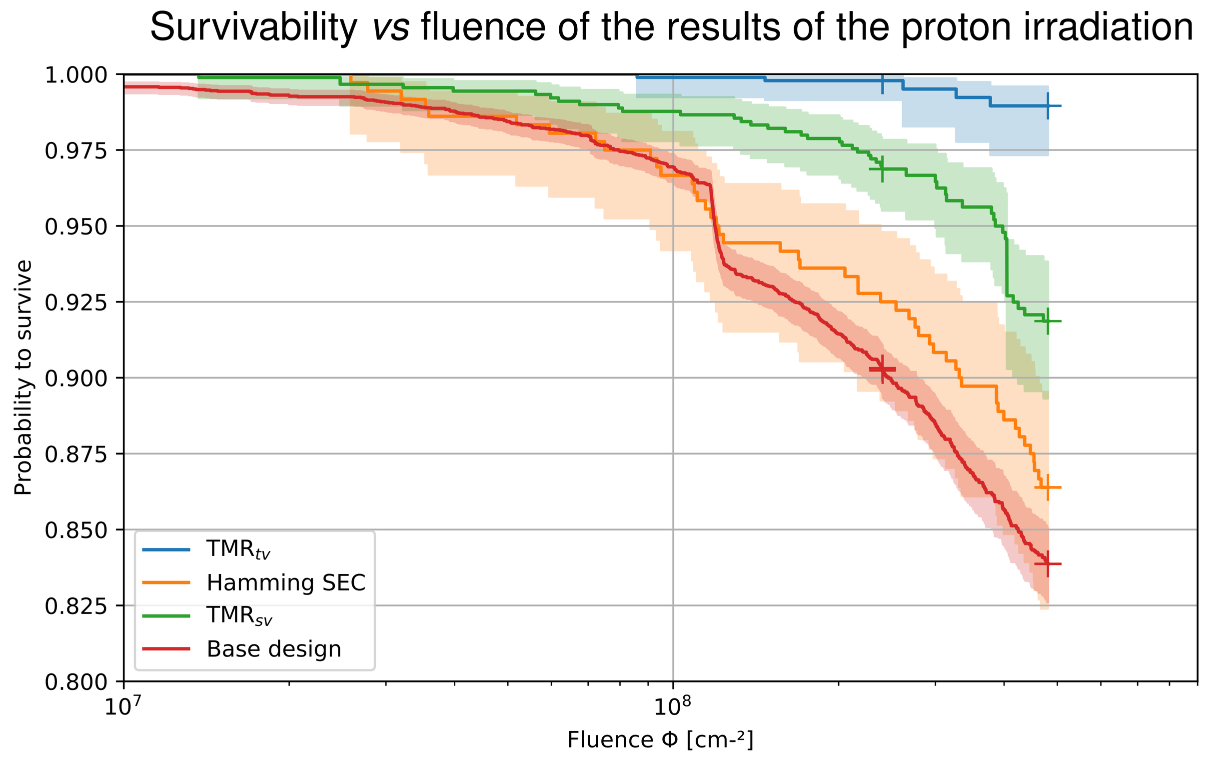



The results in the two plots (right) show a significant difference between the different radiation effect mitigation techniques. The differences between the fault injection (top) and the proton irradiation (bottom) are mainly in the different amount of statistics gathered, *i.e.* in the confidence interval of the survivability plot. The non-mitigated design has, as expected, the highest cross-section. The **Hamming with SEC** and the **TMR with a single voter** have **similar performance**: the one with the lowest resources occupancy, *i.e.* TMR is preferable. When the number of accumulated errors increases, however, they start failing since the feedback mechanism is not protected. The **TMR with a triple voter**, however, shows a **significantly better performance** than the other solutions. This is due to the higher level of protection guaranteed by the implementation, which can keep working despite one out of three voters and FSMs being affected by an error. The triple voter also guarantees that the correct value is fed back to the $\delta$ combinatorial logic.

Survivability *vs* fluence of the results of fault injection




Survivability *vs* fluence of the results of the proton irradiation




The Mean Time To Failure (MTTF) for a single FSM potentially used in the ITS readout can be calculated using the target flux at the ITS readout electronics position, *i.e.* $f = 1 \cdot 10^3\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$. The values in the table (right) are relative to the cross-sections estimated during the proton irradiation. The **TMR with triple voter** guarantees a much **higher MTBF** if compared to the other techniques. This method has a cross-section of at least one order of magnitude lower while having similar resources utilisation to the Hamming implementation. Assuming approximately 100 FSMs per RU, we can estimate that **TMR with triple voter** provides **more than a full day of MTTF**, considering the whole system. For comparison, the average data acquisition run in ALICE is approximately ten hours.

Mean fluence and time to failure as calculated from the proton irradiation results for 8-state FSMs protected with different radiation effect mitigation techniques

| Mitigation technique | MΦTF $1 \cdot 10^{12}$ [cm$^{-2}$] | MTTF (1 RU) $1 \cdot 10^4$ [day ] | MTTF (192 RUs) $1 \cdot 10^2$ [day ] |
|---|---|---|---|
| No mit. | $1.58 \pm 0.64$ | $1.83 \pm 0.75$ | $0.95 \pm 0.39$ |
| Ham. SEC | $1.67 \pm 0.92$ | $1.94 \pm 1.07$ | $1.01 \pm 0.56$ |
| TMR$_{sv}$ | $3.60 \pm 2.24$ | $4.17 \pm 2.60$ | $2.17 \pm 1.35$ |
| TMR$^a_{tv}$ | $\mathbf{26.70} \pm 2.67$ | $\mathbf{30.90} \pm 3.09$ | $\mathbf{16.10} \pm 1.61$ |

[a] Lower limit with CL $= 0.99$

## References


[1] B. Abelev et. al [ALICE Collaboration]. Technical Design Report for the Upgrade of the ALICE Inner Tracking System. Technical Report CERN-LHCC-2013-024. ALICE-TDR-017, Nov 2013.

[2] Gianluca Aglieri Rinella. The ALPIDE pixel sensor chip for the upgrade of the ALICE Inner Tracking System. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 845:583 – 587, 2017. Proceedings of the Vienna Conference on Instrumentation 2016.

[3] K. M. Sielewicz, G. A. Rinella, M. Bonora, P. Giubilato, M. Lupi, M. J. Rossewij, J. Schambach, and T. Vanat. Experimental methods and results for the evaluation of triple modular redundancy seu mitigation techniques with the xilinx kintex-7 fpga. In *2017 IEEE Radiation Effects Data Workshop (REDW)*, pages 1–7, July 2017.

[4] Daniel McMutrey and Keith S. Morgan. Estimating tmr reliability on fpga using markov models. Technical Report 149, Brigham Young University, Dec 2008.



TWEPP-19, Topical Workshop on Electronics for Particle Physics, Santiago de Compostela, Spain, 2-6 September, 2019