Università degli Studi di Napoli "Federico II"

Scuola Politecnica e delle Scienze di Base
Area Didattica di Scienze Matematiche Fisiche e Naturali

**Dipartimento di Fisica "Ettore Pancini"**

*Laurea Magistrale in Fisica*

# FPGA-based Self-repairing Digital Circuits Design for the Belle II Experiment

| **Relatore** | **Candidato** |
|---|---|
| Dott. Raffaele Giordano | Sara Massarotti |
| | Matricola N94/394 |

A. A. 2019/2020

# Contents

# Introduction

High Energy Physics (HEP) detectors produce a huge amount of data per unit time (Tbps and more), thus requiring dedicated trigger and data acquisition (TDAQ) systems with real-time processing capabilities. Belle II is an HEP experiment at the rare/precision frontier, that requires very selective on-line triggering and real-time processing of the produced data. Indeed, it is designed for the measurements in the heavy flavour sector of the Standard Model (SM) and to search for New Physics (NP). Belle II is designed to study the B mesons and weak decays for the measurement of CP symmetry violation. The Belle II detector is built at SuperKEKB at the KEK laboratory in Tsukuba, Japan. It is an $e^+ - e^-$ collider with an unprecedented design luminosity of $8 \times 10^{35}$ cm$^{-2}$s$^{-1}$. Belle II comprises both detectors and electronics, operating for the off-detector readout operations and on-detector real-time processing and data transfer. The on-detector digital electronics based on programmable devices has been increasingly being used over the last two decades. In particular, static RAM-based Field Programmable Gate Arrays (SRAM-based FPGAs) are increasingly used on-detector. They are reconfigurable, offer high-speed processing capability, embed dedicated resources for high-speed I/O and for Digital Signal Processing (DSP). Due to the SuperKEKB operation, detectors and electronics operate in a radiation environment, that may impact their functionality. The Beam Exorcism for a Stable Experiment II (BEAST II) commissioning detector has been designed to measure the radiation effects and to prevent radiation damage of detectors and electronics. It comprises a monitoring system for radiation effects on three SRAM FPGAs, selected due to their use in some Belle II sub-detectors.

The main issue with the usage of FPGAs in radiation environments are single event upsets (SEUs) in the configuration memory. These SEUs might generate an unintended behaviour of the firmware. In order to preserve the correct functionality of the circuit, several mitigation techniques for radiation-induced errors have been developed. They can be split in two main categories: redundacy- and reconfiguration-based techniques. Such methods operate at different levels. Indeed, the first group acts to mask the error effects in the circuit, whereas the second directly manipulates the configuration, restoring the original configuration of the circuit. Since masking does not prevent SEU accumulation, redundancy-based methods are advantageous when combined to the reconfiguration-based ones. Among the redundancy-based techniques, the Triple Modular Redundancy (TMR) approach is the most used, consisting in majority voting the outputs of three copies of the same circuit. Instead, the reconfiguration method, also known as the configuration scrubbing, is performed by periodically correcting the FPGA configuration. This can be performed with a variety of techniques from a simple overwrite to the usage of redundant configuration or error correcting codes

In this thesis work, I have designed and implemented a self-repairing circuit based on the redundancy of the configuration memory on a single Xilinx FPGA of BEAST II (XC7K325T). The proposed self-repairing system, named the Configuration Consistency Corrector ($C^3$), leverages both TMR and configuration scrubbing. In particular, the redundancy of the configuration allows to implement a self-repairing circuit without external memories, i.e. the resources needed to protect the FPGA firmware are all included within it. Moreover, an internal digitally-controlled-oscillator (DCO) generates the system clock to the scrubber, providing a fully self-contained system. As a starting point, I have considered the architecture designed for another FPGA and I have adapted it to the BEAST II FPGA. Furthermore, I have enhanced its functionality by modifying the software organization, and I have further facilitated the software debugging which in turn simplified the tests on the circuit functionality. In particular, the changes on the software have been aimed to enhance the versatility of the circuit, in order to improve its portability onto other BEAST II FPGAs. As far as the software debugging is concerned, my work allowed to update the program executed by the device without regenerating the circuit firmware. I have also added a software check for the protection of circuit functionality and device. Finally, I have added commands for improving the fault-injection testing of the circuit. Since the system frequency is a key parameter, I have conceived a calibration system of the internal digitally controlled oscillator to determine its best value according to the oscillator layout.

I have studied a particular technique of design flow supported by Xilinx, the Isolation Design Flow (IDF). It is conceived to optimize the isolation between different modules within the same FPGA device and enhance the reliability without using multiple device. Thus, I have modified the previous architecture in order to implement a second version of the self-repairing circuit according to the IDF requirements.

Finally, I have designed and performed dedicated tests to evaluate the self-repairing capability of the two versions. These tests aimed to simulate SEU effects on the circuits. In particular, they are able to identify the bits defining the circuit functionality and drive the circuit itself to corrupt them and attempting self-repair. Then, I have identified the configuration memory bits causing a circuit failure, i.e. which cannot be self-corrected, and I have performed a statistical analysis on their distribution for the two proposed circuits.

The outline of the thesis is the following:

- in Chapter 1, I introduce the Belle II experiment with the BEAST II commissioning detector and its FPGA monitoring system;
- in Chapter 2, I describe the radiation effects on the electronic components;
- in Chapter 3, I focus on the mitigation techniques for SEUs effects in the configuration memory of the SRAM-based FPGAs;
- in Chapter 4, I present two self-repairing circuit implementations;
- in Chapter 5, I describe and analyse the fault-injection test results on tests circuits and on the implemented self-repair circuits.

# Chapter 1

# The Belle II experiment

## 1.1 Physics motivation and overview

The Belle II experiment is an High-Energy Physics experiment at the rare/precision frontier designed for the measurements in the heavy flavour sector of the Standard Model (SM) and to search for New Physics (NP) phenomena. To reach these purposes, the Belle II experiment aims for an unprecedented target luminosity and is designed to study the B mesons weak decays. The latter have an important impact on models beyond the Standard Model (BSM). Indeed, if a charged Higgs existed [1], branching ratios in the decays $B \to \tau\nu$ and $B \to D^{*}\tau\nu$ could be modified, leading to deviations from SM predictions. The high luminosity at SuperKEKB, the collider for the Belle II experiment, allows to determine the branching ratios with unprecedented sensitivity. Furthermore, the measurement of CP symmetry violation in the B meson weak decays has been improved compared to the Belle experiment, and it is aimed to explain the matter-antimatter asymmetry in the NP scenario. CP symmetry violation observed in such processes is allowed in the SM if a nonzero phase appears in the standard parametrization of the Cabibbo-Kobayashi-Maskawa (CKM) matrix, V in the following. V successfully predict the CP violation adding a third generation of quarks, the bottom quark, to the starting non-CP-violated Cabibbo formulation of 1963. This formulation was pointed out by Kobayashi and Maskawa in 1973, that theorized the existence of the bottom quark as the only way to have a realistic model of CP violation. Indeed, such a particle was discovered in 1977 the bottom quark and its antiparticle are the main constituents of the B mesons. As a consequence, the CKM provides a flavour mixing of three quarks in weak decays, connecting the mass eigenstates $(d, s, b)$ with the corresponding weak eigenstates $(d', s', b')$:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \tag{1.1}$$

The $\mathbf{V}$ matrix is a unitary matrix fully defined by four independent real parameters: three mixing angles $\theta_{i,j}$ and the CP-violating phase $\delta$ in the standard representation. Experimental data and the unitarity condition $\mathbf{V}^\dagger \mathbf{V} = \mathbb{I}$ give an estimation of the three angles in addition to a specific hierarchy involving their sines. These conditions can be implicitly taken into account by considering the Wolfenstein parametrization [2]. The latter is derived from the observation that $\mathbf{V}$ differs from the unity $\mathbb{I}$ by a small quantity, due to the value of $V_{us}$ equal to 0.22, the measurement of B mesons lifetime yielding to $V_{cb} \simeq 0.06$, and other experimental information. Thus, by replacing $V_{us}$ by $\lambda$, $V_{cb}$ can be expressed as $A\lambda^2$ with $A \simeq 5/4$. Finally, the unitarity condition allows to write $\mathbf{V}$ in the final form to order $\lambda^3$

$$V \simeq \begin{pmatrix} 1 - \frac{\lambda^2}{2} & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \frac{\lambda^2}{2} & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & -A\lambda^2 & 1 \end{pmatrix} + \mathcal{O}(\lambda^4), \tag{1.2}$$

where $\rho$ and $\eta$ are new parameters, whose combination $\rho - i\eta$ allows a direct calculation of the CP-violating phase. The Wolfenstein representation is an approximation of the standard representation and, unlike the latter, shows parameters of the same order of magnitude.

The unitarity of the CKM matrix implies the existence of relations between the rows and the columns of the matrix itself and these relations are represented as triangles in the complex plane. The most common used triangle is described by the relation $V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0$ and is depicted in Fig. 1.1. The angles of such triangles can be calculated by measuring the CP symmetry violation, with the most relevant contribution coming from the observation of different B meson weak decay channels.

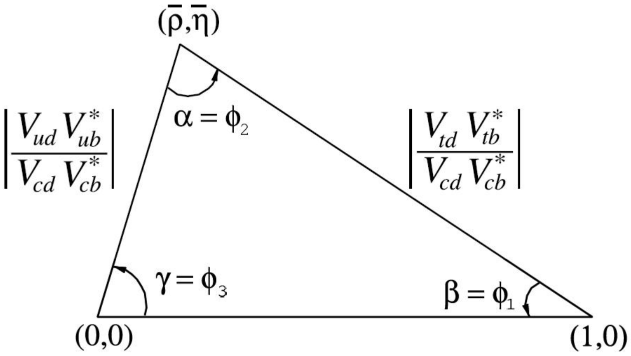

$(\bar{\rho},\bar{\eta})$
$\left|\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*}\right|$
$\alpha = \phi_2$
$\left|\frac{V_{td}V_{tb}^*}{V_{cd}V_{cb}^*}\right|$
$\gamma = \phi_3$
$\beta = \phi_1$
(0,0)
(1,0)

Figure 1.1: The Unitarity Triangle.

In the Belle II experiment, B mesons are produced in pairs $B^0\overline{B}^0$ from the decay of $\Upsilon(4S)$, that is an accessible resonance with invariant mass $m_{\Upsilon(4S)} \simeq 10.58\ GeV/c^2$ in the electron-positron collision performed by the SuperKEKB asymmetric collider. Therefore, SuperKEKB allows to realize a second-generation B-Factory in the Belle II experiment as the new generation B-Factory. The $\Upsilon(4S)$ resonance has a quark composition $b\bar{b}$ and is the lightest resonance with a mass sufficient to decay in $b$-flavoured mesons. The process $e^+e^- \to \Upsilon(4S)$ is selected by choosing an energy of the $e^+ - e^-$ system in the centre of mass frame equal to $\simeq$ 10.58 GeV, and this is convenient due to the branching ratio $\mathcal{B}(\Upsilon(4S) \to B^0\overline{B}^0) \simeq 48\%$ [3]. The B mesons pair is produced in an entangled quantum state in order to assign opposite and distinct flavours to $B^0$ and $\overline{B}^0$. The $\Upsilon(4S)$ resonance is produced boosted with respect to the laboratory frame (i.e., the detector rest frame) [4], due to the different energies of electron and positron beams. A suitable difference between the latter is needed to enhance the lifetime of B mesons and, therefore, to improve the detection of $B^0$ and $\overline{B}^0$ particles. Instead, $B^0\overline{B}^0$ pair is produced almost at rest with respect to $\Upsilon(4S)$ ($m_{\Upsilon(4S)} - 2mB^0 \simeq 19MeV$), so that in the laboratory frame B mesons have almost the same flight direction of the boost. The final Lorentz boost factor of the $B^0\overline{B}^0$ pair centre of mass frame is $\beta\gamma = 0.28$, corresponding to a B meson mean flight distance of 130 $\mu m$. This value is sufficient to track the vertices of the B mesons, but is reduced compared to KEKB ($\beta\gamma = 0.42$). To obtain the same $\beta\gamma$ of KEKB, the energy of the positron beam should be reduced (from 4 GeV to 3.5 GeV), but this implies higher beam losses not sustainable for the luminosity requirements.

In the next sections SuperKEKB collider and Belle II detector will be described, as the main parts of the Belle II experiment.

## 1.2 SuperKEKB collider

The SuperKEKB collider has been designed as the upgrade of the KEKB collider [5, 6], shut down in 2010, and is located at KEK (High-Energy Accelerator Research Organization) in Tsukuba, Japan. A key parameter improved in SuperKEKB compared to KEKB is the luminosity, measured as instantaneous and integrated luminosity. Given an event having a cross section $\sigma$ and a number of events per unit of time $dN/dt$, the instantaneous luminosity is defined as

$$\mathcal{L} = \frac{1}{\sigma}\frac{dN}{dt}. \tag{1.3}$$

Instead, the integrated luminosity is $\int \mathcal{L}dt$. Whereas KEKB collider reached an instantaneous luminosity of $2.11 \times 10^{34} cm^{-2}s^{-1}$, SuperKEKB collider has a target full peak instantaneous luminosity of $8 \times 10^{35} cm^{-2}s^{-1}$ - 40 times higher than KEKB one - reachable by 2022 and a target integrated luminosity of 50 $ab^{-1}$ by 2025. On 21th June 2020, SuperKEKB achieved the world's highest luminosity for a colliding-beam accelerator, setting a record of $2.40 \times 10^{34} cm^{-2}s^{-1}$.

The SuperKEKB accelerator is shown in Fig. 1.2 and consists of a Low Energy Ring (LER) for the positron beam, a High Energy Ring (HER) for the electron beam, and an injector linear accelerator (Linac) 600 m long supported by a 1.1 GeV positron damping ring (DR). The electrons are produced in a pre-injector by a pulsed laser directed on a cold cathode target, giving a positron bunch with a charge of 4.4 nC and emittance[1] of 10 mm mrad, and then injected in the Linac. Here, the positron beam is produced and injected in the SuperKEKB along with the electron beam.

- **Positron beam.** A high-current 10 nC electron beam is generated and directed toward a tungsten target located in the middle of the Linac. The obtained 4 nC positrons are guided through the DR in order to reduce the emittance of the beam by radiation damping [7]. Then, they are extracted by

[1] Formally, the beam emittance is defined as the phase space surface occupied by the beam. It is commonly used as a figure of merit to designate the quality of the beam spread.

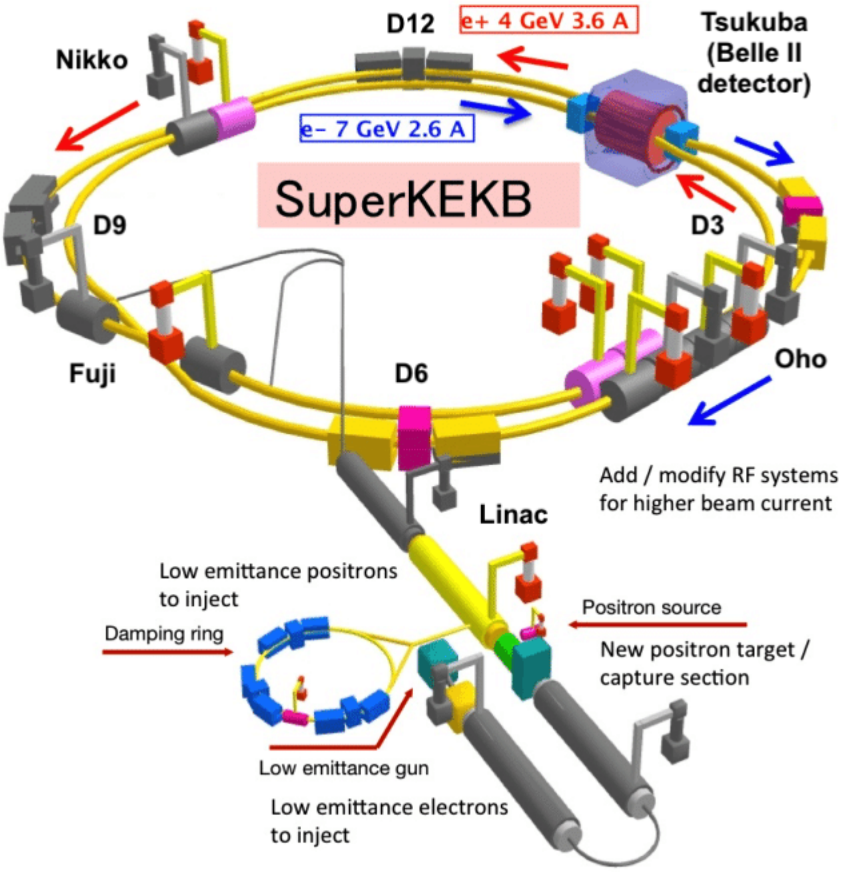


Figure 1.2: Schematic drawing of the SuperKEKB accelerator. Linac preliminarly accelerates electrons and positrons, that then enter the HER (blue arrows) and the LER (red arrows).

a capture section and accelerated up to 4 GeV in the latter half of the Linac and injected in the LER.

- **Electron beam.** The electrons are accelerated to 7 GeV at the beginning of the Linac and injected in the HER.

Hence, Linac acts as two virtual independent accelerators for the electron and positron beams, changing parameters in a pulse-by-pulse basis at 50 Hz. Particles in the beams are grouped into bunches. Each ring can store 2506 bunches with roughly $10^{11}$ particles per bunch. Newly accelerated bunches are continuously injected from the linear accelerator into the main storage rings. When new particles enter the storage ring, it takes a few milliseconds until their emittance is sufficiently reduced.

The particle beams are brought to a collision at the interaction point (IP), which is surrounded by the Belle II detector. The area around the IP extends for about 4 m along the storage rings and is called interaction region (IR). It is equipped with quadrupole magnets for the final focusing of the beams. The beampipe around the IR consists of two Beryllium layers with Paraffin in-between them acting as a coolant. The high stiffness, low atomic number, good thermal stability as well as its diamagnetic nature validate the choice of Beryllium as beampipe material. The low atomic number is advantageous to suppress multiple scattering of particles coming from the IP and the diamagnetic properties ensure that no interference with the magnetic field of the focusing magnets can occur. An additional gold coating at the inside of the beampipe is used to suppress synchrotron radiation.

The LER and HER rings are divided into four arc sections (D3, D6, D9 and D12), hosting normal conducting dipole bending magnets to guide the particles around the ring, and four straight sections (Tsukuba, Oho, Fuji and Nikko). Both bent and straight sections are supported by focusing magnets and collimators in order to maintain a small diameter of the beam. Superconducting and normal conducting radiofrequency (RF) cavities are installed along the straight sections to mitigate the energy loss due to the synchrotron radiation. The latter is induced by the bending and focusing magnets along the storage ring that accelerate the beam particles, being present especially in the HER and giving emitted photons from few keV to tens of keV.

The method allowing to achieve the high target luminosity of SuperKEKB is the nano-beams scheme, developed by the Italian physicist Pantaleo Raimondi and used for the first time in this collider. The basic idea is to squeeze the electron and positron bunch beam sizes at the IP, in both the horizontal and vertical directions of the transversal plane, in order to increase the luminosity. Indeed, the latter varies inversely with cross-sectional area of the colliding beams. The luminosity is also proportional to the product of the two beam currents ($\mathbf{L} \propto i_{e^+} i_{e^-}$), i.e., the numbers of particles colliding per unit of time. The crossing area is further reduced by increasing the crossing angle $2\phi$ of the beams up to 83 mrad (Fig. 1.3), about four times compared to the KEKB one. The crossing angle is used to avoid a head-on

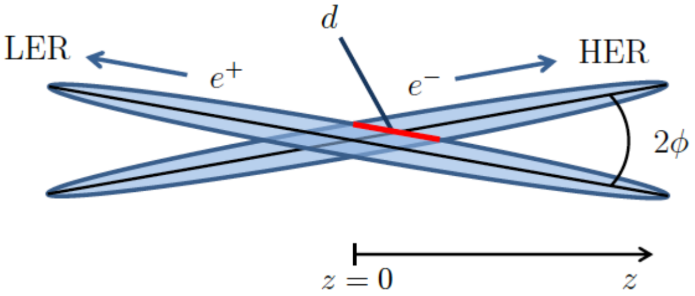


Figure 1.3: Geometry of the crossing angle.

collision, which would introduce various difficulties and disadvantages due to beam optical effects. Due to the beams squeezing at the IP and the increased crossing angle, the perturbation from the nominal trajectory of each beam is reduced by a factor 20 with respect to the KEKB case and the target luminosity can be obtained just by doubling the beams currents values of KEKB (1.64 A/1.19 A in KEKB and 3.60 A/2.62 A in SuperKEKB for electron and positron beams, respectively).

### 1.2.1 Background at Belle II

The electron-positron collisions at the Belle II IP induce different undesired background processes. The interactions of the beam particles with each other or with residual gas in the beampipe constitute the beam-induced background, whereas the high beam current values in SuperKEKB provoke the luminosity-dependent background. The main sources of these two kinds of background are briefly described below.

### 1.2.2 Beam-induced background

The first source of the beam-induced background is the Touschek effect [8], that is an intra-bunch Coulomb repulsive scattering. It occurs due to the high density of particles in a beam bunch needed for the high luminosity demand of SuperKEKB. Indeed, the larger the density of particles in a beam bunch, the higher the collision rate between particles within a bunch. Touschek effect is observed when the particle collisions have an energy so high that a particle leaves the bunch. Particles trajectories deviations also occur because of the interactions between the beam particles and the residual gas, $H_2$ and $CO_2$, contained in the beampipe, that constitute the second source of the beam-induced background. However, the particles deviating from their nominal trajectories can collide with the beam-pipe inner wall, producing a shower. If the shower develops close to the interaction region, the shower particles can enter the sensitive part of the detector. Countermeasures used for Touschek background are efficient also for beam-gas background.

The last beam-induced background source is the synchrotron radiation, since, especially in the HER, particles have an energy loss in terms of emitted photons from few keV to tens of keV, as anticipated.

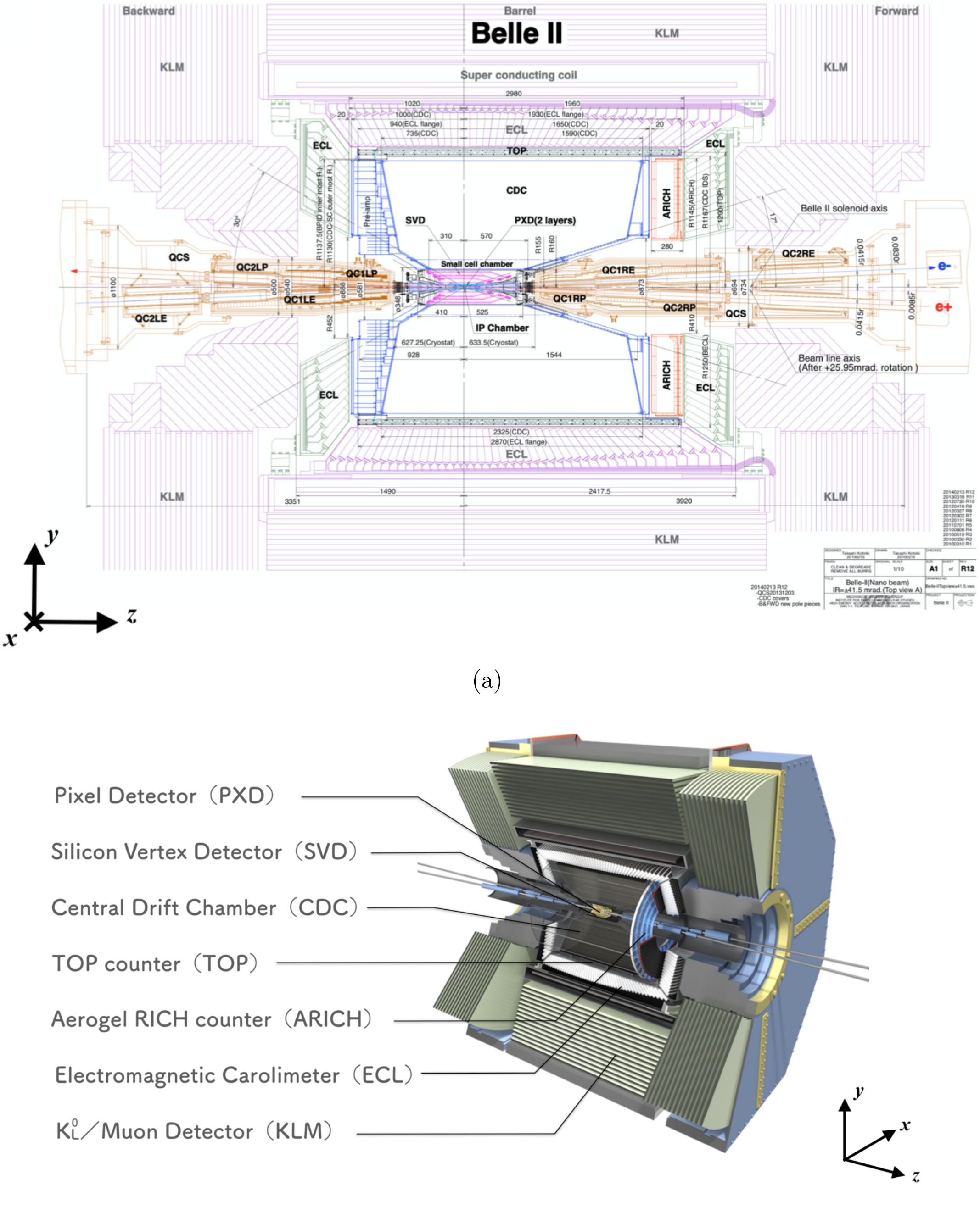
### 1.2.3 Luminosity-dependent background

The luminosity-dependent background scales with the luminosity of the accelerator [9]. The background at SuperKEKB is therefore approximately 40 times larger compared to KEKB [10]. The high beam currents in SuperKEKB enhance the probability that the radiative Bhabha scattering $e^+e^- \to e^+e^-\gamma$ occurs, having an a priori high cross section in electron-positron collisions. The scattered particles could lead to back-scattering into the detector, whereas the emitted photons could induce the emission of neutrons[2], which are able to create severe damage in the sensors and electronics of the innermost part of the Belle II detector. Another important luminosity-dependent background sources are the low momentum $e^+ - e^-$ pairs produced in the two-photon process $e^+e^- \to e^+e^-\gamma\gamma \to e^+e^-e^+e^-$, since they might spiral inside the detector.

## 1.3 The Belle II detector

The Belle II detector is the upgrade of the Belle detector and it is installed at the SuperKEKB collider IP. In Fig. 1.4 a top view and a three-dimensional view of the detector are shown. It is designed to cope with higher backgrounds related to the higher luminosities. Also, it has been optimized for the $B^0\overline{B}^0$ identification

[2] The photons produced in the electromagnetic showers hit pipe nuclei and extract neutrons.



(a)

Pixel Detector（PXD）
Silicon Vertex Detector（SVD）
Central Drift Chamber（CDC）
TOP counter（TOP）
Aerogel RICH counter（ARICH）
Electromagnetic Carolimeter（ECL）
K$_L^0$／Muon Detector（KLM）
y
x
z

(b)

Figure 1.4: The Belle II detector. (a) Detailed top view with all the sub-detectors , (b) three-dimensional sketch.

and reconstruction. Finally, the Belle II detector exhibits excellent vertexing and tracking capabilities allowing the discrimination of $B^0$ from the $\overline{B}^0$.

The Belle II detector is conventionally described by a right-handed Cartesian coordinate system with the origin located at the nominal IP. The $z$-axis is parallel to the beampipe. The positive direction of the $z$-axis points to the forward region of the detector. The positive part of the $y$-axis points to the top of the detector and the orientation of the $x$-axis is parallel to the radial direction towards the outside of the detector ring. It has an asymmetric design to account for the forward boost in the direction of the electron beam. The angular acceptance of Belle II is in the range $[17°, 150°]$ for the polar angle $\theta$, and it is $2\pi$ for the azimuthal angle $\varphi$. With regard to the polar angle, the detector is divided into three regions:

- the forward region ($17° < \theta < 30°$);
- the barrel region ($30° < \theta < 125°$);
- the backward region ($125° < \theta < 150°$).

The regions differ in terms of detector set-up and material budget. The Belle II detector is composed of several sub-detector systems, for vertex finding, tracking and identifying the decay products. The sub-detectors are described in detail below.

### 1.3.1 Vertex detector (VXD)

The vertex detector in the Belle II experiment reconstructs the B-meson decay vertices with a precision of 100 $\mu m$. The excellent position resolution improves the impact parameters when compared to Belle. It is a six-layer system forming a cylinder around the beampipe and consisting of two sub-detectors: the inner two-layer Pixel detector (PXD), mounted directly onto the beam pipe around the IP, and the outer four-layer Silicon vertex detector (SVD). The scheme of the detector is depicted in Fig. 1.5. The quality of the vertex reconstruction given by the VXD is related to different sensors constituting PXD and SVD. Due to the nano-beam scheme of SuperKEKB collider, the beampipe radius is just 10 mm allowing a good vertex reconstruction. As a consequence, the silicon vertex strip detectors, used in Belle, are no longer suitable for the innermost layers of the vertex detector because

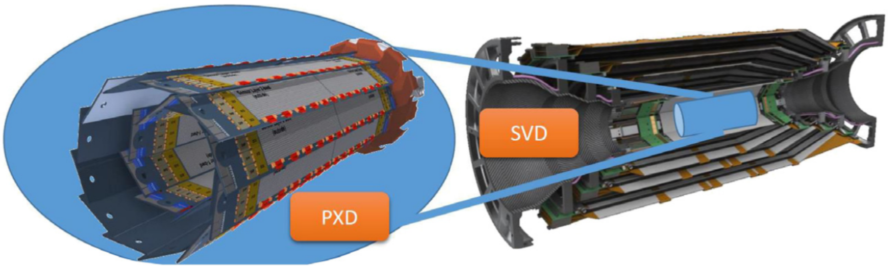


Figure 1.5: Belle II Vertex Detector concept. In the picture, a magnified PXD (in the left) is surrounded by the SVD (in the right). [11]

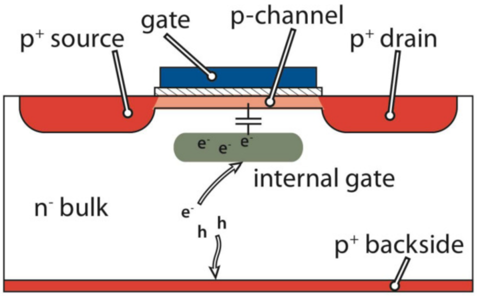


Figure 1.6: Schematic depiction of a p-channel DEPFET.

strip sensors have a large occupancy, defined as the fraction of channels hit in each triggered event. For this reason, the first two layers of VXD are composed of pixel sensors, that count a larger number of channels and therefore provide a smaller occupancy to improve the spatial resolution.

#### 1.3.1.1 Pixel detector (PXD)

The pixel sensors are distributed on two layers, at 14 mm and 22 mm from the IP, and are based on the DEPFET technology (DEPleted Field Effect Transistor) [12]. The latter allows to have granular, fast, lightweight, and very thin pixel sensors (50 $\mu m$), that are also able to withstand the high-radiation environment. The DEPFET pixels consist in a p-channel MOSFET integrated onto a fully depleted n-type silicon bulk, with a $p^+$ backside contact (Fig. 1.6). The bulk depletion generates a potential minimum, directly underneath the transition channel at a depth of about 1 $\mu m$. Incident particles generate electron-hole pairs within the fully depleted bulk. While the holes drift to the back contact, electrons are accumulated in the potential minimum, called the internal gate. When the transistor is switched on, the electrons modulate the channel current. This scheme shows various intrinsic advantages:

- reduction of white and electronic noise [3], even at room temperature, related to the small capacitance of the internal gate;
- signal amplification and direct readout, avoiding charge loss between collection and amplification;
- non-destructive readings of the stored charge, allowing for different readout schemes.

However, the charged particle tracking limits the PXD (and SVD) features mainly due to the multiple scattering. Indeed, the material budget is 0.2% $X_0$ per layer.

#### 1.3.1.2 Silicon vertex detector (SVD)

The SVD detector is composed of four layers Double-sided Silicon Strip Detectors (DSSDs), placed at 38 mm, 80 mm, 115 mm, and 140 mm from IP. The average radii range between 3.8 cm (layer 3) and 13.5 cm (layer 6), covers 90% of the solid angle. The SVD detector employs several kinds of sensors differing by shapes and strip pitch. Furthermore, it is designed to optimize the resolution of the impact parameter and to sustain the background condition expected at full luminosity. As for the PXD, the material budget is limited, having a value of 1-2% $X_0$ per layer in this case. In addition, due to the increase of a factor 40 in luminosity, the background-hit rate will be sensibly higher than the Belle B-Factory. Finally, the occupancy is kept under control by shorter strips.

### 1.3.2 Central Drift Chamber (CDC)

The CDC is a large volume He-$C_2H_6$ 50-50 drift chamber with three important purposes: trajectories measurement, momenta measurement, and energy loss of charged

[3] Here, the electronic noise is referred to the 1/f or pink noise, where f indicates the signal frequency.

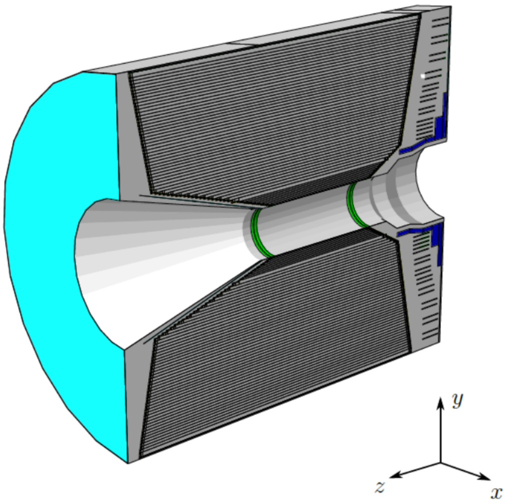


Figure 1.7: Schematic three-dimensional drawing of the CDC structure.

particles within its gas volume, for charged particles with momenta ranged from 50 MeV/c to 8 GeV/c. It contains 14336 wires arranged in 56 axial (parallel to the beampipe) or stereo (slanted with respect to the axial wires) layers, having the schematic structure reported in Fig. 1.7. Compared to the Belle CDC, Belle II CDC counts six more layers and almost a double number of sense wires, justified from the larger radii of the inner and outer cylinders and wires. Low-momentum tracks, which do not reach the subsequent particle identification system, can be identified using the CDC alone. Finally, the central drift chamber, gives reliable and efficient trigger signals for charged particles.

### 1.3.3 Particle Identification System (PID)

The particle identification (PID) system in the Belle II detector is needed to distinguish of charged particles such as pions, kaons, protons, electrons, and muons deriving from the $B^0$ decays. This identification is performed by the ring-imaging Čerenkov detectors located in the barrel region and in the forward end-cap region, namely the Time of Propagation Counter (TOP) and the Aerogel Ring-Imaging Čerenkov counter (ARICH), respectively. Čerenkov detectors measure the $\theta_c$ angle of photons emitted by relativistic charged particles crossing a radiator material, ob-

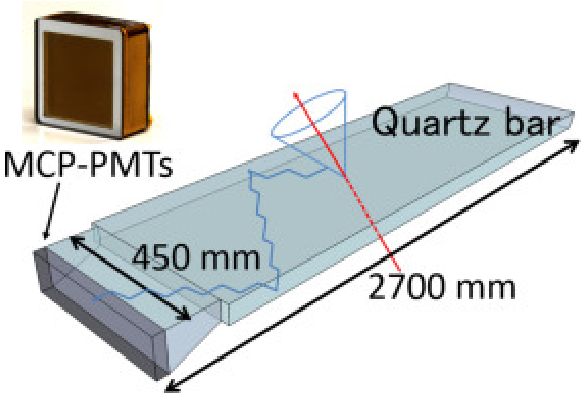


Figure 1.8: Single module of the TOP counter. The charged particles cross the quartz radiator bar and the emitted photons reach the MCP-PMTs by total reflections, also supported by the focusing mirror at the forward end of the bar.

taining $\beta$ of the particle with the following relation $\cos\theta_c = (\beta n)^{-1}$ [4], where $n$ is the refractive index of material.

#### 1.3.3.1 TOP

The TOP counter is composed by 16 identical modules, that are arranged around the IP in a barrel-like geometry. As shown in Fig. 1.8, each module is composed of a quartz radiator bar, a focusing mirror at the forward end of the bar, and an array of micro-channel plate photomultipliers tubes (MCP-PMTs) at the backward end of the bar. Due to the high average refractive index of the quartz, the Čerenkov radiation produced by the radiator remains trapped by total reflection, propagating to the MCP-PMT array. Thanks to the pixel size of $\simeq 5.5 \times 5.5\ mm^2$ and a transit time spread less than 50 ps, the MCP-PMTs provide a very precise measurement of position and detection time. Once the direction of the incoming particle is known, the time of propagation of the Čerenkov photons inside the quartz is only function of the Čerenkov angle. Therefore, the TOP provides a combined measurement of both time of flight $t_{TOP}$ and Čerenkov angle. Overall, the TOP counter has a single photon time resolution of about 100 ps.

[4] This relation derives from the distance covered by the Čerenkov radiation $ct/n = \beta ct\cos\theta_c$, where $\beta c$ is the charged particle velocity in the material and $c/n$ is the radiation velocity emitted at $\theta_c$ with respect to the particle direction.

#### 1.3.3.2 ARICH

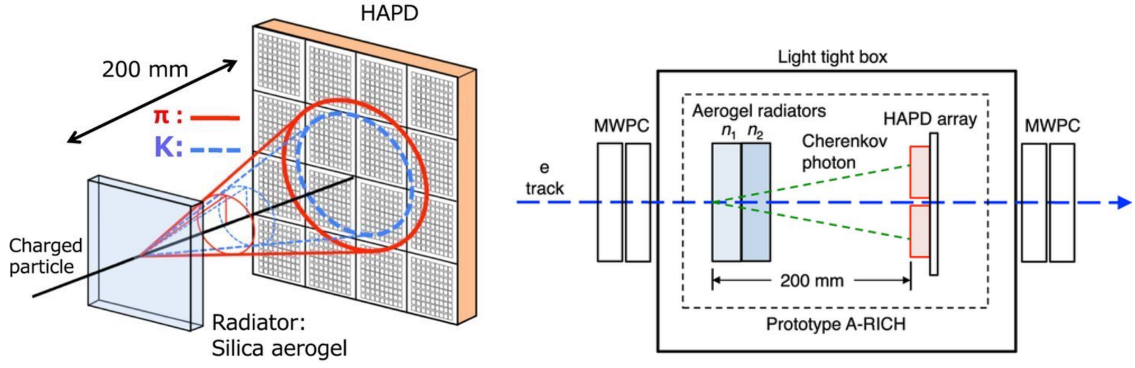
A fundamental requirement of the Belle II experiment is the identification of the charged particles over the full kinematic range of the experiment, ranging from a few tens of MeV/c to ≈8 GeV/c. In the forward end-cap, the subdetector achieves this goal. It is a proximity focusing Ring-Imaging detector schematized in Fig. 1.9 and having the structure described below.

- Aerogel tiles work as radiator to produce Čerenkov photons starting from incident charged particles.
- An expansion volume of 20 cm allows the Čerenkov photons to enlarge into rings on the surface of an hybrid avalanche photon detector (HAPD).
- The HAPD detects single photons, having the Čerenkov angle $\theta_c$.
- A read-out system collects electrical signals from the HAPDS.

The key parameter in the performance of a RICH counter is the Čerenkov angle resolution per track $\sigma_{track} = \sigma_{\theta_c}/\sqrt{N_\gamma}$, where $\sigma_{\theta_c}$ is the single photon Čerenkov angle resolution and $N_\gamma$ is the detected photons number. With a longer radiator, $N_\gamma$ increases but $\sigma_{\theta_c}$ degrades because of the emission point uncertainty. The peculiar solution for Belle II ARICH is using two layers of aerogel with different refractive indices ($n_1 = 1.045$, $n_2 = 1.055$) as a radiator of 2 cm thickness, so that the $N_\gamma$ value is equivalent to a double radiator thickness and $\sigma_{\theta_c}$ reaches an optimized value of $\simeq$ 13 mrad for charged tracks with momentum larger than 3.5 GeV/c, even if no significant degradations are shown for lower momentum tracks. Therefore, with $N_\gamma \simeq 10$ per tile, the Čerenkov angle resolution per track is $\sigma_{track} \simeq 3$ mrad. The angular acceptance is $\theta \in [14°, 30°]$.

### 1.3.4 Electromagnetic calorimeter (ECL)

The employment of a high resolution electromagnetic calorimeter (ECL) in B-decays detector is of utmost importance because one third of the decay products are $\pi^0$'s or other neutral particles like $K^0$'s that provide photons in a wide enery range from 20 MeV to 4 GeV. The Belle II ECL uses the same scintillation crystal material, namely



(a) (b)

Figure 1.9: The ARICH counter. (a) The principle of $\pi$ and $K$ identification. (b) The complete scheme for the Belle II ARICH by using two layers of aerogel with different refractive indices.

CSI(Tl) (thallium-doped cesium iodide), but is supported by a total upgrade of the read-out electronics to cope with the SuperKEKB increased luminosity. In addition to the photons energy and angle measurement, it contributes to the $K^0$ detection with $K_L$ and $\mu$ system (KLM) detector.

The ECL comprises a barrel region, a forward end-cap, and a backward end-cap region, covering 90% of the solid angle in the centre of mass frame with an angular acceptance of $\theta \in [12.01°, 155.03°]$.

### 1.3.5 Superconducting coil

The superconducting coil produces an homogeneous magnetic field $\vec{B}$ of 1.5 T, parallel to the beam direction. The coil is made of NbTi/Cu, carrying a current of 4400 A and operating under a liquid helium cryogenic system. It encloses an internal cylindrical volume of radius $r = 1.7$ m and length of 4.4 m. The iron structure of the KLM works as the return yoke of the magnetic field to confine the latter. Therefore, in the region of the KLM outside the coil, the direction of $\vec{B}$ is inverted.

### 1.3.6 $K_L$ and $\mu$ system

The KLM task is to identify the muon tracks by measuring their penetration depth and to reconstruct the $K_L^0$ by considering the combined information of ECL and the hadronic KLM showers. It is formed by alternating 4.7 cm iron plates and scintillators with silicon photomultiplier (SiPM) detectors surrounding the coil. As anticipated, the iron plates provide the magnetic flux return for the coil. Furthermore, they provide 3.9 interaction lengths ($\lambda_0$) in addition to the $0.8\lambda_0$ of the ECL, in which $K_L^0$ can shower hadronically. Octagonal barrel covers the polar angle range from 45° to 125°, whereas backward and forward end-caps extend this coverage from 20° to 155°. The barrel has 15 detector layers and 14 iron plates, whereas each end-cap has 14 detector layers and 14 iron plates. The barrel layers are Resistive Plate Chambers (RPCs):

- thoroughgoing charged particle ionizes the gas molecules along its path in a proportional gas chamber;
- an electric field accelerates the electrons toward the anode and the ions toward the cathode;
- a strong electric field enables a streamer mode, i.e., a significant enhancement of ionizations between the electrodes;
- dielectric foam surrounds the electrodes to prevent the propagation of sparks;
- the signal is read by means of metallic strips on one side of the chamber.

Compared to Belle detector, in the end-caps and the innermost layers in the barrel region the RPCs have been replaced by layers of scintillator strips with a read out by silicon photomultipliers (SiPMs). This upgrade is due to the long dead time of the RPCs, that significantly reduces the detection efficiency for the high background hit rates present in the Belle II detector. Moreover, this would give a large muon misidentification probability. On the contrary, the layers of scintillator strips have an high rates tolerance, although the large neutron background degrades the SiPMs and, therefore, considerably increases their dark count rate. Nevertheless, such a detector system can be reliably operated by appropriately shifting the discrimination threshold. The muons are identified starting from CDC tracks: each track is extrapolated to KLM region with a $\pi$ mass hypothesis, and a KLM hit is assigned to that track if is present near the extrapolation region. A $K_L^0$ can be reconstructed clustering all of the KLM hits within a 5° opening angle con from the IP, and then applying a charged track veto by means of a CDC track matching. If the remaining neutral KLM clusters are aligned within a cone of 15° with an ECL cluster, the two showers are associated. The $K_L^0$ detection efficiency rises linearly from 0 at 0 GeV/c to a 80% plateau at 3 GeV/c. The angular resolution is about 3° for KLM-only candidates and 1.5° for KLM+ECL candidates. The SiPMs offer an excellent time resolution of $\sigma_t \simeq 0.7$ ns, allowing to also measure the $K_L^0$ time of flight.

## 1.4 Monitoring radiation effects in detectors and electronics

Once the sub-detectors structure in the Belle II detector has been described, the practical implementation issues related to the radiation effects must be discussed. Usually, detectors and electronics employed in the High-Energy Physics (HEP) experiments work in radiation environments, exposing the system to possible damage. The BEAST II (Beam Exorcism for a Stable Experiment II) is a commissioning detector designed to measure these effects, making it possible to prevent radiation damage on the sub-detectors and tuning models for background simulation. In the next subsections, the BEAST II structure and the monitoring system of specific electronic components, namely Field Programmable Gate Arrays (FPGAs), will be described. The concepts related to the radiation damage and their mitigation in the electronic integrated circuits, will be more deeply treated in the next chapters.

### 1.4.1 The BEAST II: SuperKEKB Commissioning Detector

The BEAST II is a SuperKEKB commissioning detector, needed to prevent radiation damage to the Belle II detector and to improve simulations of beam-induced backgrounds near the IP. Indeed, the experience with KEKB has shown that the measurement of the particle and X-ray backgrounds around the IP is crucial to ensure the radiation to be sufficiently low before the vertex detector is installed. During the Phase 1 of the Belle II data taking[5], BEAST II consisted of eight detector systems described below [13].

- In 32 points near the IP, there are 64 PIN diodes, not biased to simplify the associated electronics, which provide the monitoring of the ionization radiation dose. Ionizing radiation leaves a free electrons and holes trail with a consequent enhancement of the dark current from the PIN diodes. The integral of this amplified current is proportional to the ionizing radiation dose.

- A system made of single-crystal diamond detectors performs the monitoring of the instantaneous and integrated radiation doses. This system is designed to preserve the vertex detector, due to the their exposition to the largest radiation doses. The readout electronics system offers continuous monitoring of radiation doses and also aborts the signals related to beam losses corresponding to an excessive levels of luminosity.

- An inorganic scintillator electromagnetic calorimeter, namely "Crystals", measures the electromagnetic background radiation in the innermost part of Belle II ECL. As in the latter, similar position and detection technology is employed to simulate the observed measurements. Indeed, Crystals has an accurate time resolution for bunch-by-bunch beam-induced backgrounds and for the injected backgrounds relative to the Touschek and beam-gas contributes. The Crystals system is composed of six identical units, each containing three crystals read-out by means PMTs. The three crystal types are CsI(Tl), pure caesium iodide, CsI(pure), and cerium-doped lutetium yttrium orthosilicate, LYSO. The CsI and LYSO crystals operate in a fast read-out mode to measure the time structure of injection backgrounds.

- A detector system consisting in a bismuth germanium oxide (BGO) monitors the real-time beam backgrounds as electrons and gammas. Furthermore, the BGO monitors the luminosity of the collider as well, by counting Bhabha

[5] Phase 1 consisted in the SuperKEKB commissioning to characterize the beam environment and from Feb. to June 2016.

events rate for focused beam. In the Belle II experiment, eight BGO crystals sensors are installed around the IP: four in the forward region and four in the backward region, with light-tight treatments applied to the BGO in order to ensure maximum light-collection efficiency and to prevent leakage of environmental light.

- A system composed by eight plastic scintillators tiles read-out with SiPMs, namely CLAWS, measures background levels connected to injection with time resolution higher than the bunch crossing frequency (250 MHz). The CLAWS measures the total rate and the exact time arrival of minimum-ionizing particles (MIPs). Finally, the CLAWS are sensitive to MeV neutrons.
- $^3$He detector system consisting of $^3$He tube system measures the rate of thermal neutrons (kinetic energy of $\simeq 0.25eV$). Each detector is a stainless steel cylinder with 5.08 cm in diameter and 20.38 cm in length, filled with $^3$He and a small amount of $CO_2$ at a pressure of 4 atm. In the centre of the tube there is a sense wire, which is set to a voltage of 1.58 kV. The $^3$He tubes are located above, below, and on either side of the IP.
- A system of four Time Projection Chambers (TPC) provides the direction and the energy measurements of fast neutron recoils produced by the various beam backgrounds. The TPCs are located around the IP at $\varphi = 0°$, 90°, 180°, and 270°. The TPCs provide detailed 3D measurements of charge density distributions via micro pattern gas detectors.

### 1.4.2 System for Monitoring Radiation Effects in Field Programmable Gate Arrays in BEAST II

As anticipated, the main purpose of BEAST II is monitoring the effects of the radiation environment surroundings the detector. The radiation impact is linked to both detector systems, for the vertex sub-detector protection, and the electronic components. Among the latter, the static RAM-based FPGAs (SRAM-based FPGAs) are widely used in trigger and data acquisition system of Belle II detector, due to their re-configurability and large data processing and transfer capabilities.

For this reason, three test boards hosting Xilinx FPGAs are included in BEAST II. Two boards host Kintex-7 (70T and 325T), one hosts Virtex-5 LX50T FPGAs. The specific FPGA families have been chosen basing on their use in two Belle II sub-detectors:

- Virtex-5, since it is installed the CDC for read-out data;
- 7-series, since it is installed on the TOP counter

Virtex-5 LX50T and Kintex-7 325T are installed on the forward side of the detector at about 10 m from the IP along $z$-axis, whereas Kintex-7 70T is installed on the backward side of the IP at about 7 m. The different installation positions are needed to investigate the inhomogeneous radiation environments created by the asymmetric colliding. In the radiation tolerance field, Single Event Effects (SEE), mainly Single Event Upsets (SEUs) in the configuration memory, affect and damage the FPGAs. Although the scaling of technological processing allows high total dose hardness levels in the FPGAs, the development of SEU impact mitigation techniques are needed and they will be deeply examined in the Chapter 3 whereas the radiation damage in the electronic component will be discussed in the next Chapter.

# Chapter 2

# Radiation Damage in Electronic Components

In HEP, silicon-based technology is widely used for detectors and readout systems. In this chapter the radiation effects on the electronic components will be discussed, referring to silicon unless differently indicated.

## 2.1 Overview of radiation effects in electronic components

The matter-radiation interaction has a key role in the functionality of electronic devices. Indeed, it could induce damage essentially as alteration of the crystal structures or undesired production of charge carriers, giving macroscopic effects in the circuits behaviour. Therefore, the radiation effects can be split in two categories based on the radiation-damage mechanism and described below.

- **Ionization damage**. The ionization effect can occur as cumulative effect - related to the concept of total ionizing dose (TID) - or as single event effect (SEE). The cumulative effect occurs when the energy absorbed in a semiconductor or insulating layer produces free charge carriers, i.e., electron-hole pairs. The latter diffuse or drift to other locations, being trapped in some cases. However, the consequent charge accumulation leads to a parasitic field. This effect is the primary one in X-rays, $\gamma$-rays exposition, and charged particles, and it is important in devices based on surface conduction as the metal-oxide-silicon field effect transistors (MOSFETs). SEEs are related to the charge deposition induced by a single particle crossing a sensitive region of the device.

- **Displacement damage**. Incident radiation displaces atoms from their lattice sites, altering the structure and the electrical characteristics of the crystal. Therefore, displacement damage are also cumulative effects and mostly affect bulk properties of devices. Their main consequences on the electronic devices are trapping phenomena and the generation of in-gap states.

Firstly all, radiation damage can be fully understood and defined if the information on the incident radiation are known. Among these, there are the particle type ($e$, $p$, $n$, $\gamma$, ions, etc.) and energy, the fluence $\Phi$ or flux $\phi$[1], the chances for certain effects occurring (cross section and threshold), and the knowledge of the stochastic or predictable character of the effects. Also, the effects of the radiation depend on the properties of target, such as the chemical composition, the device type (p–n junction, MOSFET, bipolar junction transistor (BJT), etc.), their active volumes and sensitivity. At typical energies involved in HEP radiation environments, ionization is the dominant absorption mechanism and primarily depends on the absorbed energy, independently on the radiation type. Ionization damage is proportional to the energy per unit of mass (dose), usually expressed by gray, 1 Gy = 1 J/kg, or rad, 1 rad $= 6.24 \times 10^{16}$ eV/kg (1 Gy = 100 rad). Ionization effects depend on to the target material, since the produced charge by a given dose depends on its electronic properties and crystal structure. Therefore, the ionizing dose must be referred to a specific absorber.

Displacement damage depends on the non-ionizing energy loss (NIEL) [14], i.e., the energy and momentum transferred to the lattice atoms, which exhibits a strong dependency on both particle type and energy. For this reason, the measure of displacement damage must be based on specific particle type and energy.

[1] The flux $\phi$ is defined as the number of particles per unit area and time and the fluence $\Phi$ is defined as the number of particles per unit area. These two parameters are important in the measure of radiation damage.

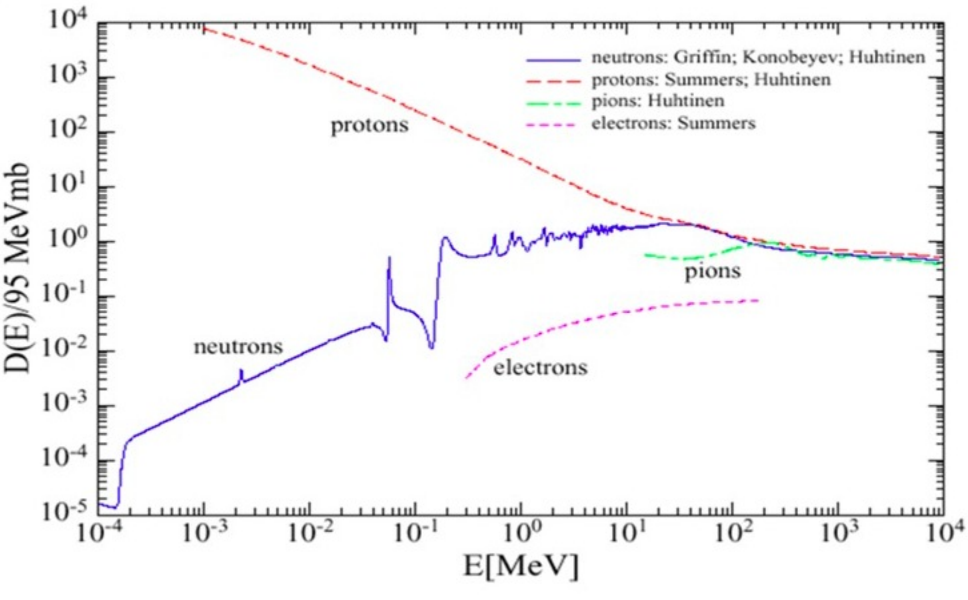


Figure 2.1: Displacement damage in Si as a function of the energy of neutrons, protons, pions, and electrons, normalized to 1 MeV neutrons. [15]

## 2.2 Total dose-based effects

Firstly, the total dose-based effects will be described. In particular, the main focus will be on phenomena induced by the matter-radiation interaction in MOSFETs and diodes.

### 2.2.1 Displacement damage

Displacement damage is a complex effect involving a volume or cluster of material atoms rather than a single atom. The extension of the induced defect cluster depends on the incident particles type and energy. For instance, a 1 MeV neutron transfers about 60 − 70 keV to a Si recoil atom, which in turn displaces roughly 1000 additional atoms in a region of about 0.1 $\mu$m size. For a variety of particles, NIEL has been calculated over a large energy range. Fig. 2.1 shows the displacement damage integrated in the cross section in Si as a function of the energy for neutrons, protons, pions, and electrons, plotted relative to 1 MeV neutrons. The damage evidently depends on the particle type, with the largest values corresponding to protons at smaller energies than 1 MeV neutrons. Above energy of 300 MeV, the displacement effect for protons, neutrons, and pions is almost the same. However, nor these data neither the induced cluster size and distribution are easy to predict. For example, for 1 MeV neutrons the initial vacancy distribution is highly clustered, whereas for 10 MeV protons the distribution is quite uniformly distributed, and 24 GeV protons form a mixture of clustered and uniformly distributed damage sites [16]. Moreover, the NIEL model allowed to test the radiation resistance of Si compared to other candidates for electronic devices, such as GaAs. The latter is more radiation resistant than Si, except for proton irradiation. Furthermore, it intrinsically shows defects, due to the interface effects of an heterostructure, thus exhibiting several in-gap states favouring trapping phenomena.

The displacement damage can be mainly observed through three effects in the semiconductors, which are listed in the following.

- **Formation of in-gap states**. The presence of in-gap states favours the electron transition from the valence band to the conduction band, inducing a current enhancement in the reverse-biased *pn*-junctions and a charge loss in forward biased junctions due to the favoured recombination of charge carries.
- **Trapping and detrapping phenomena**. The crystal point defects generate (flat) in-gap electronic states, where some electrons or holes can be attracted getting a localized behaviour. Therefore, trapping and subsequent detrapping effects reduce the system charge mobility or conductivity [17].
- **Alteration of the material doping characteristics**. The change in donor or acceptor density modifies the doping characteristics of the material, leading to different functionality thresholds.

The silicon band structure exhibits an indirect band-gap[2], so that an inter-band transition without momentum transfer is unlikely.

As anticipated, the in-gap states induce emission and capture processes from the defects. A full understanding of these processes can be achieved by formally defining the ground state of the defected semiconductor. The former consists in fully occupied

[2] In the Brillouin zone, the valence band maximum and conduction band minimum are related to different Bloch momenta.

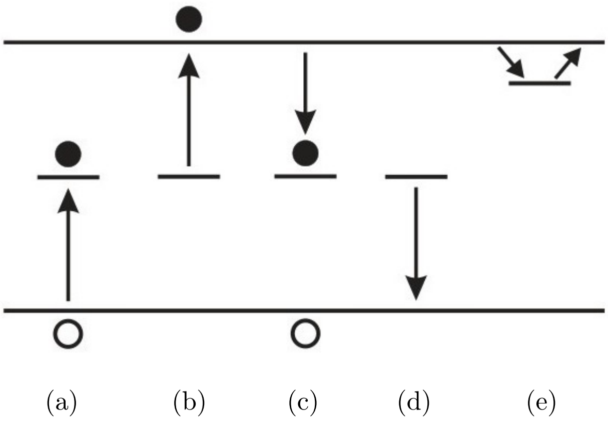


Figure 2.2: Emission and capture processes from a defect level in a semiconducting system involving electrons (filled circles) and holes (unfilled circles). (a) Hole emission, (b) electron emission, (c) electron capture, (d) hole capture, (e) trapping and detrapping. The lower and upper continuous lines represent the valence band maximum and the conduction band minimum, respectively. For each case, the in-gap segment represents the defect energy level. The lack of circles in (d) and (e) indicate that after the process the defect level is unoccupied.

valence bands and a free defect level. In Fig. 2.2, all emission and capture processes from a defect level are reported. Fig. 2.2(a) shows the hole emission process, where an hole is created in the valence band because of the absorption of an electron in the defect level[3] The electron emission process can be only the transition of an electron from the defect level to the conduction band (see Fig. 2.2(b)), since a transition to the valence band would provide a hole capture process. The electron capture process, shown in Fig. 2.2(c), occurs when a conduction band electron is captured in the defect level. The hole capture (see Fig. 2.2(d)) is the transition of the defect level electron to the valence band. Apart from these single-transition processes, the trapping-detrapping phenomenon is a double-transition process occurring when the defect level is sufficiently close to a band edge. In Fig. 2.2(e), the electron trapping is depicted, where a conduction band electron is captured by the defect level and released after a certain time, defined as the trapping relaxation time.

[3] Notice that the hole does not move from the defect level to the valence band, since the hole does make physical sense only in the valence band. Here, the emission process for the hole must be intended as a process firstly involving an electron, as well as the capture process will be.

As far as the occurrence of the emission and capture processes is regarded, there are different factors to consider. Firstly, in general, the electronic transition probabilities have the exponential form $e^{-(E_f - E_i)/k_B T}$, where $E_i$ ($E_f$) is the energy of the initial (final) state, $k_B$ is the Boltzmann factor, and $T$ is the temperature. As a consequence, the existence of electronic (defect) states between the valence band maximum and the conduction band minimum favour the electronic transitions. Also, one should take into account the distance of the defect levels from the band edges, in addition to the distribution of electronic states. Instead, the dominance of generation or recombination process is related to the concentration of carriers and to the empty defect states. In the depletion region of a reverse biased *pn*-junction, the conduction band is underpopulated and the generation prevails. On the contrary, in a forward biased *pn*-junction, the conduction band is largely populated and, thus, the recombination is the prevalent process.

The trapping phenomenon is often observed in radiation detectors and photodiode systems, where the latter imply the decrease of reverse-bias current related to the signal. Such a process decreases the carrier lifetime and, as a consequence, causes a loss of signal, due to the recombination of charge carriers while they traverse the depletion region. The defect states can act as donors, acceptors, or can be electrically neutral. The predominant charge states formed in Si are acceptor-like, and when in sufficient concentration affect the net space charge in the active region [15]. The space charge determines the voltage required for the full charge collection.

#### 2.2.1.1 Displacement damage in diodes

In HEP, reverse-biased diodes with large depletion depths ($\sim 10^2$ $\mu$m) are employed as radiation detectors and photodiodes. The features of the depletion region are strictly related to the bulk properties of the device. Therefore, in reverse-biased diodes, the displacement damage represent the main damage mechanisms, affecting their doping characteristics and influencing the detector leakage current and the charge collection.

The microscopical description of the physical phenomena occurring in the device are quite complex, due to the many processes involved. On the other hand, a macroscopic representation of these phenomena just requires simple parametrizations.

The main parameters affected by the displacement damage are the leakage current and the doping characteristics.

The formation of in-gap states in the diode causes an increase of the leakage current. The current after the irradiation depends on the particle fluence $\Phi$, the particle type and the active device volume $V = Ad$, where $A$ and $d$ are the diode area and thickness, respectively. Assuming a spatially uniform formation of active defects in the device volume regardless the details of energy levels and states, the reverse bias current after the irradiation is expressed as

$$I_d = I_0 + \alpha \Phi A d, \tag{2.1}$$

where $I_0$ is the bias current before irradiation. The factor $\alpha$ is a damage coefficient expressing the particle type dependency. For instance, the $\alpha$ value is $3 \times 10^{17}$ A/cm for 650 MeV protons and $4 \times 10^{17}$ A/cm for 1 MeV neutrons.

The correct functionality of a diode predicts a current saturation well below the depletion voltage[4]. Instead, in a radiation-damaged diode, the uniform distribution of damage sites provides an initial increase of the leakage current, approximately varying with the square root of the voltage. Afterwards, the current behaviour depends on the annealing process. The latter is a complex process involving system defects, induced by the irradiation and related to the system temperature. The annealing process modulates the migration of radiation-induced defects in semiconducting devices and allows to decrease the leakage current with time after initial defect formation. Furthermore, the reverse bias current is strongly dependent on temperature. As expected, the higher the temperature, the higher the generation current, where the latter is the electrical current generated by the emitted carriers, i.e. the generation current. In particular, even after rather low fluences the generation current dominates and the reverse bias current is

$$I_R \propto T^2 e^{-E/2k_B T}, \tag{2.2}$$

where $E \approx E_g$, depending on the impurity or defect energy level compared to the band gap value $E_g$.

[4]. The depletion voltage is normally defined as the bias voltage required so that region depleted of free carriers reaches through the whole of the semiconductor bulk REF

As far as the doping characteristic is concerned, in a reverse-biased diode, the $n$-type region exhibits a positive space charge. After irradiation, the creation of new acceptor states neutralizes the donor states and the positive space charge decreases. A certain value of the fluence implies the absence of the space charge because the new acceptor states balance the old donor states. Beyond this fluence value, a negative space charge is observed because the acceptor states result dominant. Although the diode functionality is unchanged, the bias voltage must be enhanced proportionally to the space charge increase. Indeed, the $n$-type region in a damaged device with $p$-like space charge is not equivalent to a $p$-doped material, since the space charge change does not generate mobile holes. Therefore, the change in space charge corresponds to a change in doping level, and the net space charge is an effective doping level $N_{eff}$. As a consequence, the needed bias voltage is

$$V = \frac{e}{2\epsilon} \left| N_{eff} \right| d^2, \tag{2.3}$$

where $\epsilon$ is the semiconductor dielectric constant. In analogy to the conventional diode operation, this is often referred to as the “depletion voltage”, although the device is devoid of mobile charge even at smaller voltages [15]. To estimate the value of $N_{eff}$, it is needed to consider the contributes of four processes, where the first two represent the “stable damage” contributes because their behaviour is only function of fluence, whereas the other two processes also exhibit time and temperature dependence.

- **Donor removal**. The contribute of donor removal process exponentially depends on fluence.

- **Build-up of stable charge**. The contribute of acceptor formation is a linear function of the fluence.

- **Beneficial annealing**. The contribute of beneficial annealing describes a recovery from the change in the space charge as an exponential decay of acceptors.

- **Reverse annealing**. The contribute of reverse annealing describes the increase of acceptor-like sites due to the degrade of macroscopic sensor properties and is a long-term effect.

As far as the carriers lifetime is concerned, radiation-induced traps contributes to a decrease in the lifetime: $\tau = \frac{1}{v_{th}(N_{t0}\sigma_0+(N_{tr}\sigma_r)}$. Here, $\sigma_0$ and $N_{t0}$ refer to traps before irradiation, whereas $\sigma_r$ and $N_{tr}$ refer to the new traps. The concentration of radiation-induced traps is proportional to the fluence, so that, denoting $\tau_i$ as the initial lifetime, the lifetime expression is

$$1/\tau = \frac{1}{\tau_i} + \frac{\Phi}{K}, \tag{2.4}$$

where $K$ is a constant depending on the type of captured charge and the incident radiation. Since the damage term dominates at HEP common fluences, the relaxation time is $\tau \approx K/\Phi$.

### 2.2.2 Total ionizing dose effects in devices

The ionizing radiation induces the formation of electron-hole pairs in the silicon bulk, due to the promotion of valence band electrons to the conduction bands and the consequent creation of the holes. Different effects impact on the additional charges, leading to ionization damage, that are mainly evident in systems having silicon/silicon dioxide interfaces as MOSFETs. Indeed, the ionization energy of silicon dioxide, $SiO_2$, is 18 eV[5], so that at the typical radiation energies the excitation process occurs even without momentum transfer[6] [18].

The ionization damage can be quantified through:

- interface trapped charge;
- oxide trapped charge;
- the mobility of trapped charge;
- the time and voltage dependence of charge states.

In general, the variation of the electric field with time affects the charge state of the traps and, thus, the mobility of charges. In particular, in non-equilibrium conditions

[5] Here, the polymorph $\alpha$-$SiO_2$, also known as $\alpha$-quartz, is considered, since it is the only stable form under normal conditions.

[6] The polymorph $\alpha$-$SiO_2$ shows an indirect band gap with the value ranging from 8.29 to 9.55 eV.

the local quasi-Fermi level directly sets the charge movement in the traps, in addition to the specific trap states relaxation times. As charge states also anneal, ionization effects depend not only on the dose, but also on the dose rate, in general.

The electronic devices based on Silicon technology clearly showing the ionization damage are the MOSFETs which present Si/SiO$_2$ interfaces. Therefore, in what follows, the investigation of the ionization damage will be only focused on MOSFETs, also covering the reversed-biased diodes analysis.

#### 2.2.2.1 TID effects in MOSFETs

In a $n$-channel MOSFET (NMOS), the creation of conducting channel occurs when $V_{GS} > V_{th}$ and $V_{DS} > 0$. For $V_{DS} < V_{GS} - V_{th}$, NMOS works in the ohmic region and the current flowing linearly increases with the $V_{DS}$ applied (see Fig.2.3(b)):

$$I_{D(lin)} = \mu_n C_{ox} \frac{W}{L}[2(V_{GS} - V_{th})V_{DS} - V_{DS}^2], \tag{2.5}$$

where $\mu_n$ is the electron mobility, $C_{ox}$ is the gate oxide capacitance, $W$ and $L$ are gate width and gate length, respectively. By increasing $V_{DS}$ voltage up to $V_{GS} - V_{th}$, a current saturation is observed due to the “pinch off” of the channel. As a consequence, additional increase of $V_{DS}$ value has no effect on the flowing current and the NMOS works in the saturation region (see Fig.2.3(b)):

$$I_{D(sat)} = \mu_n C_{ox} \frac{W}{L}[(V_{GS} - V_{th})^2] \tag{2.6}$$

When the radiation impacts on a MOSFET creates electron-hole pairs, drifting under the electric field effect. Due to their mobility in SiO$_2$ ($\mu_n = 20$ cm V$^{-1}$ s$^{-1}$), the electrons rapidly move towards the positive biased gate. Instead, slower holes ($\mu_h = 10^{-11} - 10^{-4}$cm V$^{-1}$ s$^{-1}$) move towards the Si-SiO$_2$ interface and, among them, a certain fraction is trapped in the oxide and the remaining ones recombine with the electrons from the bulk. Indeed, the latter have an enough high probability for tunnelling from silicon bulk and recombine with trapped holes, in a region extending $2.5 - 5$ nm into the oxide. As a consequence, to estimate the trapping probability, a crucial device characteristic is the oxide thickness. For instance, the number of holes available for trapping linearly increases with oxide thickness and the transit time. The trapping process causes a shift in the gate voltage due to the charge build-up

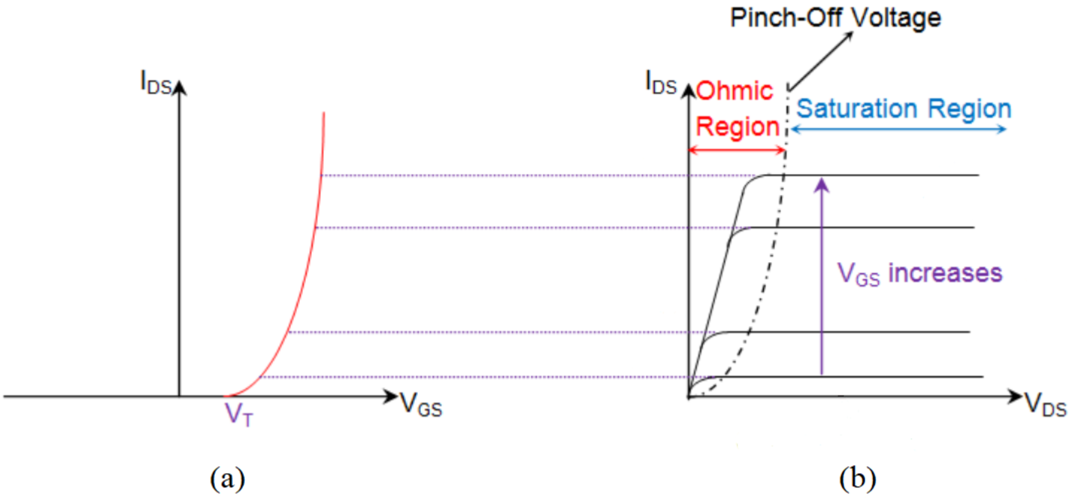


Figure 2.3: (a) Tranfer characteristic. (b) Output characteristic

and is typically expressed in terms of threshold voltage $V_T$, needed to maintain a certain current flowing. The trapped holes more distant from the Si-SiO$_2$ interface recombine with primary electrons drifting through the oxide and with hot electrons injected from the channel. The gate voltage shift is a complex function of dose rate and temperature:

- the field distribution in the oxide, governing the carrier motion and the trap population, depends on the trapped charge;
- the temperature influences the release of trapped holes and the migration of trapping sites from the Si-SiO$_2$ interface into the bulk.

In NMOS, the threshold voltage can increase, owing to the carriers drift caused by the positive gate bias voltage, or decrease, when the oxide trapped charge effect dominates, as shown in Fig. 2.4. Therefore, the $V_T$ shift can be described as sum of two contributes expressing the increase of the charge and the trapping, $Q_{ot}$ and $Q_{it}$, respectively [19].

NMOS devices are more vulnerable with respect of $p$-channel MOSFET (PMOS), where the voltage shift is monotonic, as shown in Fig. 2.5. The different vulnerability is due to the greater electron mobility, which provides more current flow in NMOS compared to the PMOS. As a consequence, with the same irradiated area, the NMOS

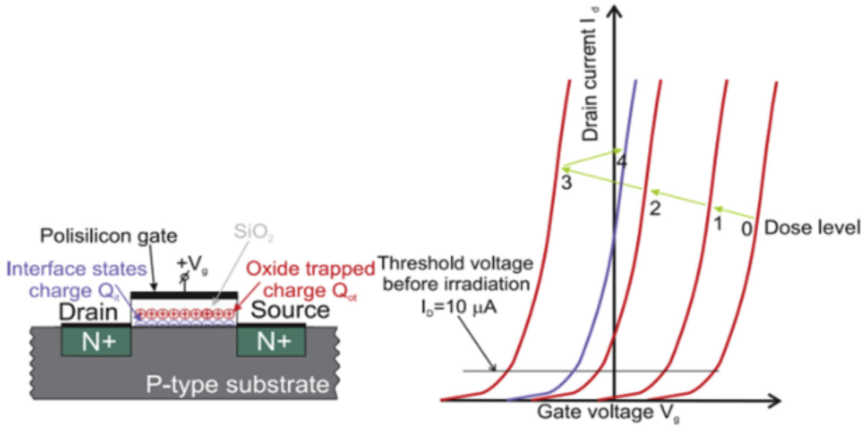


Figure 2.4: Trapped charge in a NMOS device (left) and $I_D - V_G$ curves at different dose levels (right). [20]

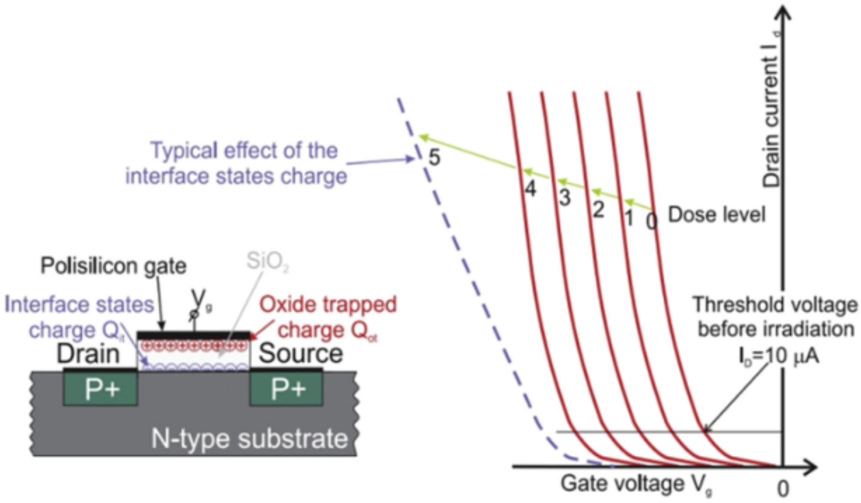


Figure 2.5: Trapped charge in a PMOS device (left) and $I_D - V_G$ curves at different dose levels (right). [21]

exhibits a percentage of area damaged two or three times greater than PMOS. Moreover, $V_T$ shift in NMOS depends on the ionization dose and can be negative or positive with consequent increase or decrease, respectively, of the leakage current. In PMOS, the $V_T$ shift is positive and, consequently, the leakage current decreases. [22]. The $V_T$ shift affects the leakage current, in turn affecting other parameters, such as the transconductance[7], the drain-source breakdown, the increase of noise, and the reduction in surface mobility. Hence, the IV characteristic shift modifies the operating point in the analogue circuits and the switching time in the digital ones.

The effects induced by the radiation damage survive for a finite time in electronic devices, such as MOS structures. After that, the system tends to reach an equilibrium state also involving parameters related to the damage. This process is known as “annealing”, and sho

The annealing process in MOSFETs involves complex mechanisms related to the interfaces and different materials characterizing the total structure. Among them, the holes detrapping from the Si-SiO$_2$ interface constitutes the main contribution to the long-term annealing of the radiation damage occurring near room temperature. In general, the holes move away from the traps via recombination with the electrons transferred from the bulk. This can occur in two ways: tunnelling and thermal diffusion. The tunneling model assumes the recombination between the tunnelled silicon electrons and the holes distribution near the Si-SiO$_2$ interface. It correctly predicts the slow bias-dependent recovery of the threshold voltage variation at normal operating temperatures (between -55° and 125°). Obviously, the recombination depth depends on the tunnelling barrier height and, in general, the time. As far as the thermal recombination, the normal operation temperatures give enough kinetic energy to SiO$_2$ valence electrons, allowing the recombination with the trapped holes. As expected, this phenomenon is favoured as long as the energy difference between traps levels and valence band is small and the temperature is high.

[7] The transconductance in MOSFET is defined as $g_m = \partial i_D / \partial V_{GS}$. For enhancement NMOS $g_m = 2k(V_{GS} - V_T)$ in saturation region.

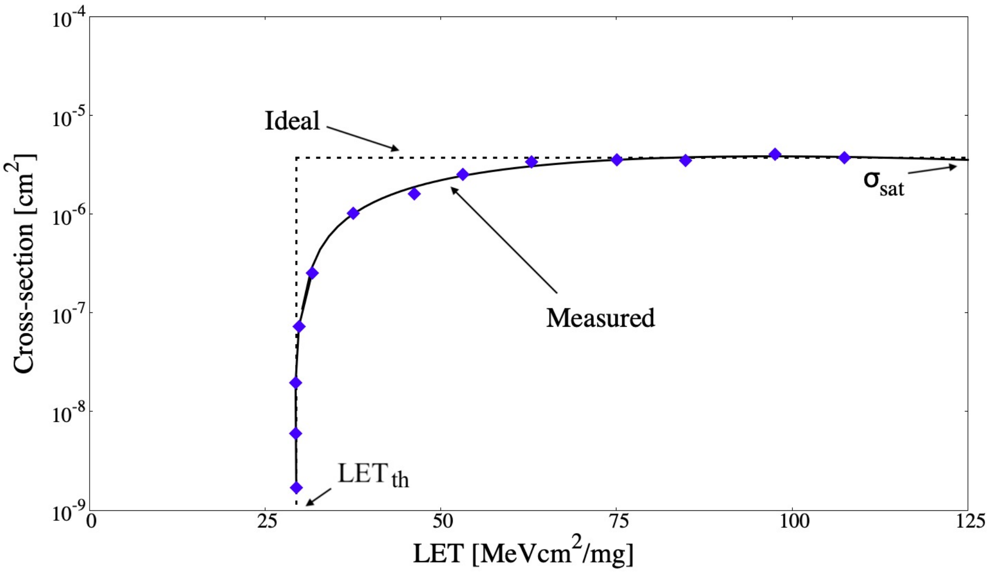


Figure 2.6: SEE cross-section as a function of LET.

## 2.3 Single event effects

Single event effects (SEEs) are due to ionization induced by a single particle crossing a sensitive device area. The amount of energy transfered by radiation is the linear energy transfer (LET), and it is measured in MeV/$\mu$m or in MeVcm$^2$/g when it is normalized to the density of the absorbing material [23]. The critical LET or LET threshold (LET$_{th}$) is defined as the minimum LET value causing a failure. Starting from this value, the cross-section reaches a saturation value $\sigma_{sat}$ by following the fit function

$$\sigma(LET) = \sigma_{sat}\left[1 - e^{-\left(\frac{LET-LET_{th}}{W}\right)^S}\right], \tag{2.7}$$

where $W$ and $S$ are fit parameters, as shown in 2.6

SEEs mainly affect technologies based on MOSFET devices, as complementary MOSFET (CMOS)[8], double-diffused MOSFET (DMOS)[9], and the integrated circuits, such as SRAM or memories based on MOSFET devices. In particular, the

[8] The CMOS is a MOSFET-based device using complementary of $p$-type and $n$-type MOSFETs for logic functions.

[9] The DMOS devices constitute a subcategory of power MOSFET, which are MOSFETs designed to hold relevant power levels in order to provide high commutation speed, along with good efficient at low power levels. The DMOSs have a vertical structure differing from the LDMOSs, which have a planar structure.

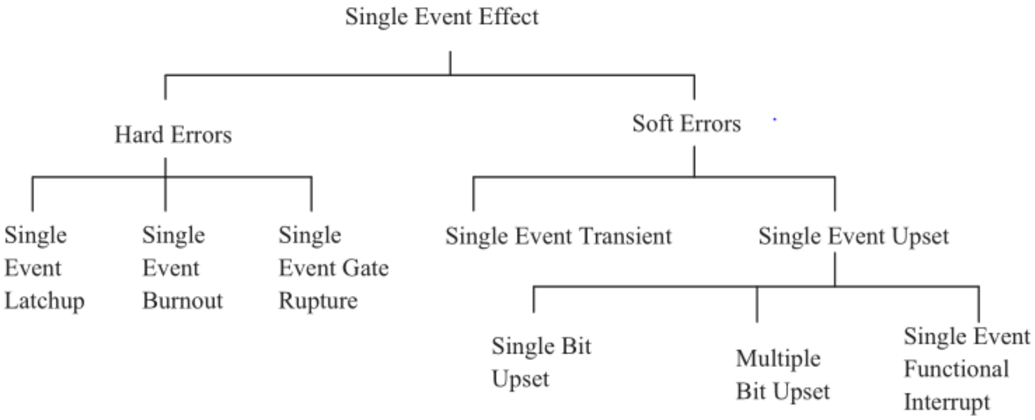


Figure 2.7: Classification of SEEs.

effect on the MOSFET behaviour is related to the reverse-biased junction formed between drain and substrate.

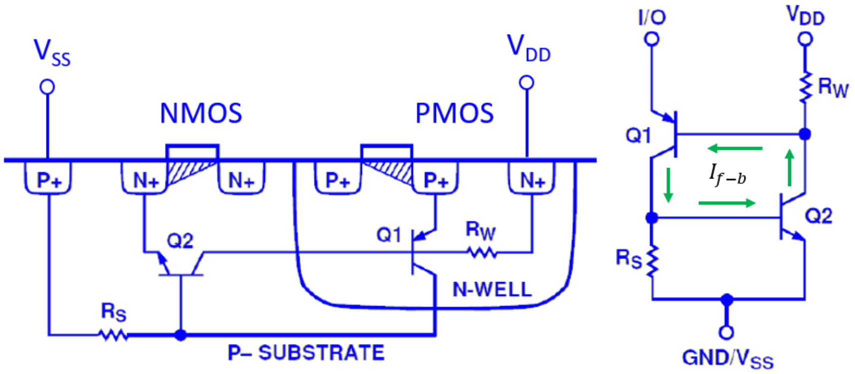
The SEE cover both phenomena classified as “soft errors”, leading to non-destructive effects, and “hard-errors”, leading to destructive effects (see Fig. 2.7). Some of the most common soft errors are:

- Single Event Upset (SEU) in SRAM memories;
- Single Event Transient (SET) in digital circuits.

Some of typical hard-errors which can be found in modern technology are:

- Single Event latch-up (SEL) in CMOS technologies;
- Single Event Burnout (SEB) in DMOS transistors;
- Single Event gate rupture (SEGR) in DMOS transistors

Read-out systems used in the Belle II experiment are mostly based on CMOS technology. Therefore, only SEEs affecting CMOS devices will be described. The destructive effects are unrecoverable damage which permanently compromise the hardware, as a burnout resulting from a short circuit [22], whereas the non-destructive effects cause damage which can be corrected by proper restoring operations.



(a)

(b)

Figure 2.8: Parasitic PNPN structure in CMOS (a) The $n$-well/$p$-substrate junction is the base collector junction of both PNP and NPN junction. The emitter-base junctions are $n^+$-source/$p$-substrate and $p^+$-source/$n$-well junctions. (b) The PNPN parasitic structure represented through two bipolar transistors.

## 2.3.1 Hard errors

### 2.3.1.1 Single event latch-up

SEL occurs when the ionizing particle activates a parasitic PNPN [10] structure embedded in the CMOS (see Fig. 2.8). This structure can be described by two cross-coupled bipolar transistors (one PNP and one NPN). An ideal PNPN structure is active when the current gain $\beta$[11] product of the NPN and PNP transistors is $\beta_1\beta_2 \geq 1$. PNPN acts as a bistable device when is active: the OFF-state arises when both transistors are OFF, and the ON-state holds when both transistors are switched-on. The current gain of BJT is related to the base width and the injection efficiency of the emitter-base junction. The base width varies with design rules and the position of the source relative to the well-substrate junction[12]. The latch-up can switch the PNPN structure from high impedance to a low impedance state. The trigger mechanism can be illustrated by considering a scheme with cross coupled NPN and PNP bipolar junction transistor (BJT), where the resistors are in series with emitters and between base and emitters. In the ON-state, the positive feedback

[10] This structure in named thyristor

[11] The current gain is defined as $\beta = I_c/I_b$.

[12] A well is in creating region in the substrate with the opposite doping.

causes high current draw in the transistors. As a consequence, the latch-up effect can cause:

- a permanent fault when the current density exceeds a certain threshold current [22];
- a temporary malfunction recoverable after a power cycle, if the event is somehow limited

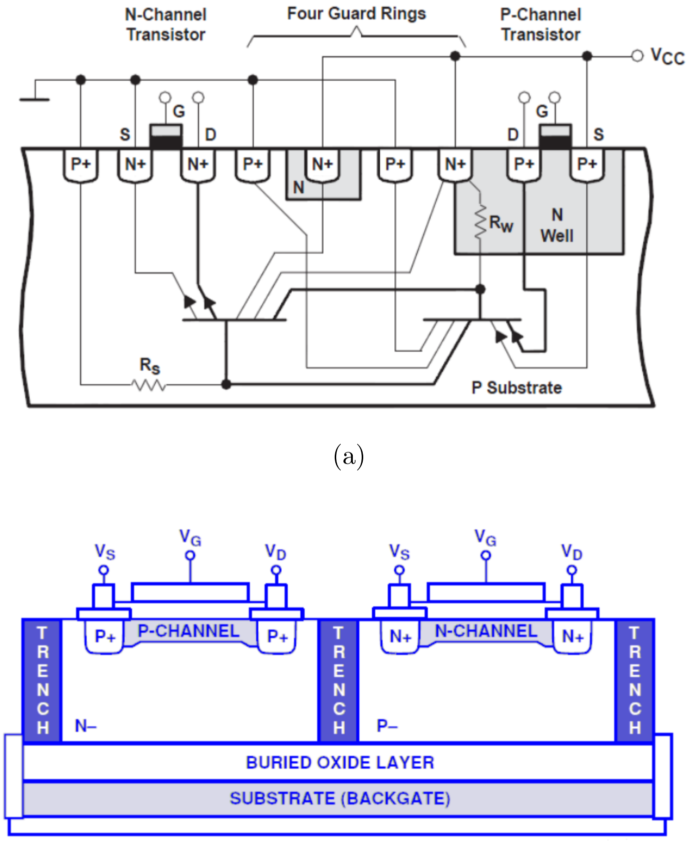
The minimum voltage and current needed to sustain the ON-state are called holding current and voltage, respectively [24]. The probability of SEL occurrence can be reduced by lowering the power voltage. The PNPN structure is in the ON-state if both base-emitter junctions and collector-base junctions are ON. In these conditions, the minimum holding voltage should be $V_{dd} - V_{ss} \simeq 0.6$ V. Actually, voltage drops in series resistors increase this minimum value, that may reach $0.8 - 1$ V. For this reason, a reduction of the power supply voltage limits the SEL occurrence. Among the other mechanisms to limit SEL phenomenon, there are (see Fig. 2.9):

- the use of guard rings around the device, which adds more collector terminals to the parasitic transistors and allows to steer the current flow away from the device (see Fig. 2.9(a));
- the isolation of NMOS and PMOS devices, using an oxide trench together with a buried oxide layer as depicted in Fig. 2.9(b).

## 2.3.2 Soft errors

### 2.3.2.1 Single event transient

The SET is a charge injection generated by a ionizing-particle strike, usually ions. Ions can impact the device directly or being generated in it as secondaries by nuclear reactions induced by the primary particles. SET can manifest itself as a glitch in the circuit. This damage mostly affects the MOSFET devices and the damage mechanism can be understood by considering the inverse biased *pn*-junction drain-substrate formed in a *n*-channel MOSFET in OFF state. The impacting ion can



(a)

(b)

Figure 2.9: Prevention SEL techniques. (a) Guard rings method. (b) Oxide trench method.

generate electron-hole pairs, affecting the electric field and, thus, inducing a current between source and drain:

- the potential barrier between source and drain is removed;
- an electric field is introduced between source and drain, causing electrons to drift;
- the consequent current between the two terminals induces a short circuit and the MOSFET behaves as in the ON-state;
- the probability of a SEU occurring increases with incidence angle.

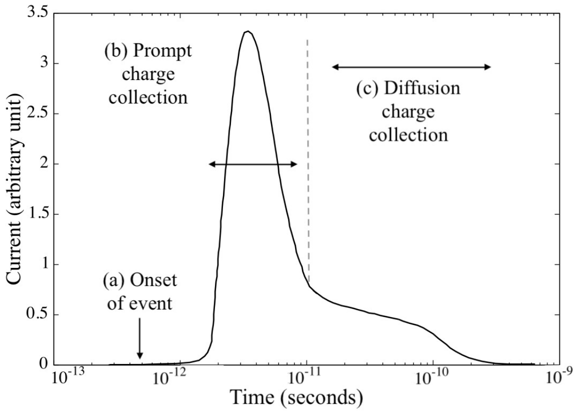


Figure 2.10: SET effect is a current peak as a function of the elapsed time.

The transition from the OFF-state to the conducting one is known as Ion-triggered-channeling (ITC). The main effect of this phenomenon is an increase of the collected charge in the drain electrode. The ITC only occurs in MOSFET devices, having less than 1 $\mu$m channel lengths. To establish if a SET is connected to a circuit error one needs to consider the capability of the transient signal to propagate through the circuit and the time relation between the SET arrival and the active edge of the memory element, in order to latch the signal. The probability that the SET is captured as valid data linearly increases with the circuit speed [24], which is determined by the clock frequency[13] (see Fig. 2.10). In asynchronous circuits, SET captures cannot be predicted through a static timing analysis. Moreover, a SET can propagate through subsequent gates and may cause a SEU if it reaches memory elements, as depicted in Fig. 2.11.

### 2.3.2.2 Single event upset

The SEU is described through the same phenomena inducing SET but it affects memory elements and consist in the change of state of a bistable element.

The SEU occurs when the signal induced by ionization particle produces a charge collection greater than the critical charge $Q_c$ needed to flip the memory cell state. The progressive MOSFET technology scale down results in a decrease of the critical charge with a consequent increase of the SEU vulnerability (see Fig. 2.12). However,

[13] High speed means faster gates and/or less logic levels per pipeline stage.

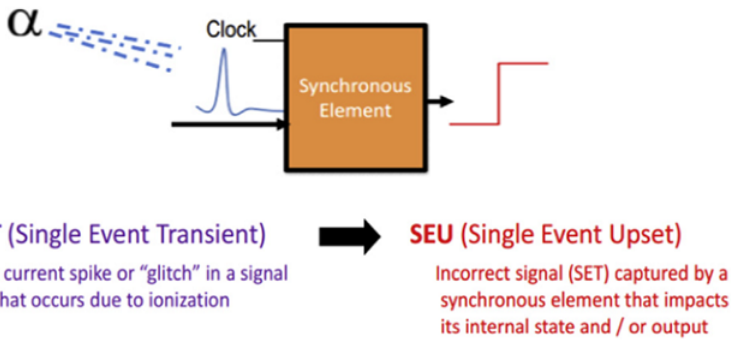
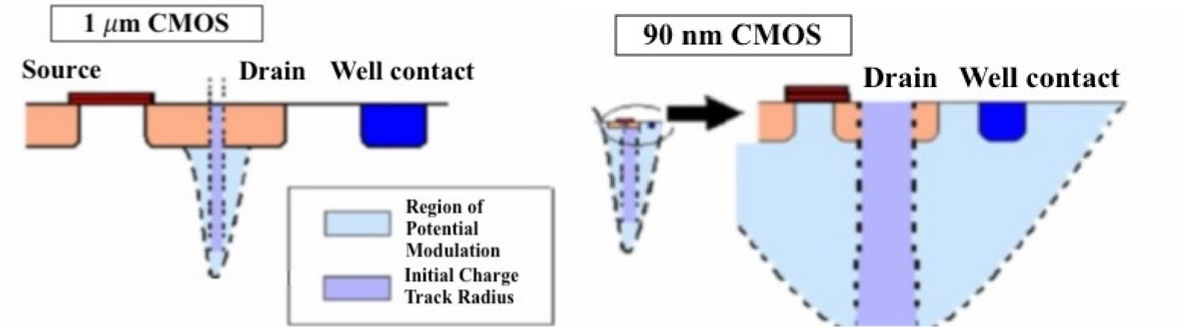

α
Clock
Synchronous Element
SET (Single Event Transient)
A current spike or "glitch" in a signal that occurs due to ionization
SEU (Single Event Upset)
Incorrect signal (SET) captured by a synchronous element that impacts its internal state and / or output

Figure 2.11: SET capture in a synchronous element, eventually causing a SEU. [22]



(a)

(b)

Figure 2.12: Schematic of SEUs occurring in CMOSs at different technology dimensions.(a) Effect of ion impact on 1 $\mu$m CMOS. (b) Effect of ion impact on 90 nm CMOS.

the latter is not a linear function of $Q_c$, due to the compensation related to the ion path influencing the deposited charge in the device. Therefore, the probability of a SEU occurring mainly depends on the LET and the particle strike location.

For instance, in 0.5 mm technology, the critical charge required to cause a SEU is roughly in the range of femtocoulombs [22]. The SEU sensitivity is measured by cross-section and is expressed in cm$^2$/bits or cm$^2$/device. The SEU effect in a six transistor (6T) SRAM cell is shown in Fig. 2.13. The cross-coupled inverters are highly asymmetric due to the stronger drive strength in the pull-down NMOS compared to the PMOS pull-ups. The asymmetry of this design causes the different speed at which the SRAM can switch the states. The NMOS constituting the second inverter is brought in conduction because of the SEU impact, which also change the

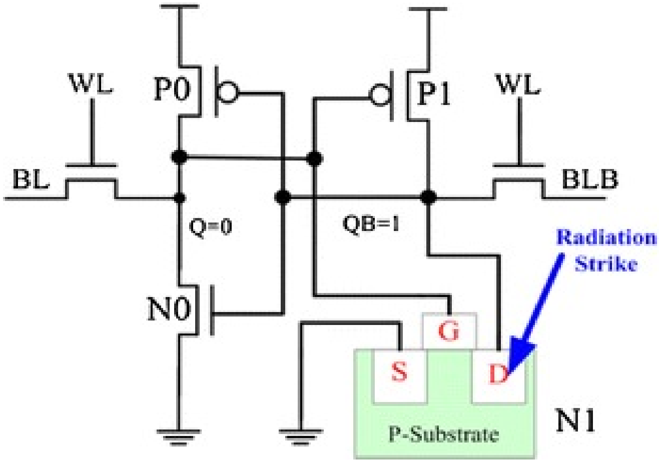


Figure 2.13: A SEU occurring in a NMOS transistor constituting a 6T SRAM cell.

value to store.

As shown in Fig. 2.7, SEUs can be split in three subcategories, which are described below.

- **Single bit upset**. A single bit upset (SBU) occurs when the particle impacting acts on a single cell and modifies the logic state of a single memory element.

- **Multiple bit upset**. A multiple bit upset (MBU) occurs when a single particle strike passes through multiple adjacent cells, affecting multiple memory cells. The MBU production can be evaluated by considering the particle impact angle and the size of the volume in which the charge is deposited.

- **Single event functional interrupt (SEFI)**. SEFI is a SEU occurring in control logic, and causes the interruption of normal device functionality.For instance, SEFI in SRAM-based FPGAs is due to upsets in particular circuits that involve power-on-reset, failures in the joint test action group (JTAG) port or select-MAP communications port, loss of configuration capability, or other effects [25, 26].

In the next chapter, the SEU impact on SRAM FPGAs and the developed mitigation techniques will be discussed.

# Chapter 3

# Single Event Effects and Mitigation Techniques in FPGA

In this chapter the soft error impact - SEU especially - and SRAM-based mitigation techniques for FPGAs, will be discussed. In particular, Xilinx SRAM FPGAs will be considered, unless differently indicated.

## 3.1 Field Programmable Gate Arrays structure

FPGAs are semiconductor devices based on a fabric of configurable logic and interconnect. After manufacturing, they can be configured according to the desired application. The FPGAs offer several programmable resources, which are described below. In order to make the discussion more practical we will refer to 7-series devices. The architecture of these FPGAs is described below and is depicted in Fig. 3.1.

- **Configuration logic blocks (CLBs).** A CLB includes two slices and a switch matrix (see Fig. 3.2(a)). A slice is composed by look-up tables (LUTs) to perform combinational operations, Flip-Flop (FF) as sequential elements, carry chains and multiplexers as depicted in Fig. 3.2(b). The switch matrix is a system of blocks that are integrated to route signals between multiple inputs and multiple outputs. Hence, this component permits communication between slices in a CLB or between other elements of FPGA and is represented in Fig. 3.3.

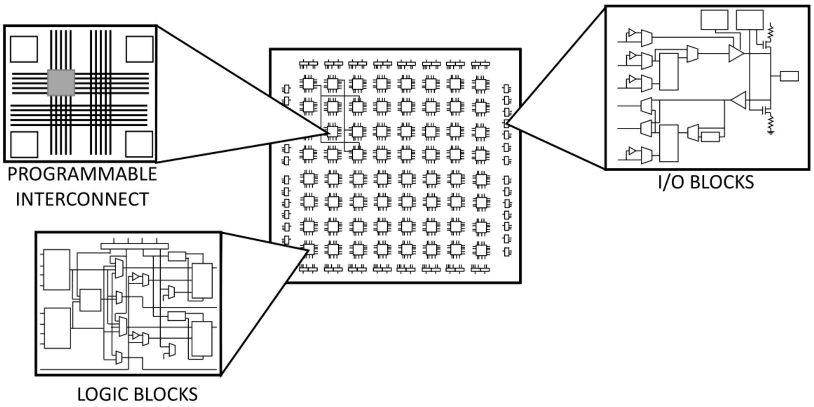


Figure 3.1: FPGA architecture representation. Details about CLBs, IOBs and programmable interconnects structure.

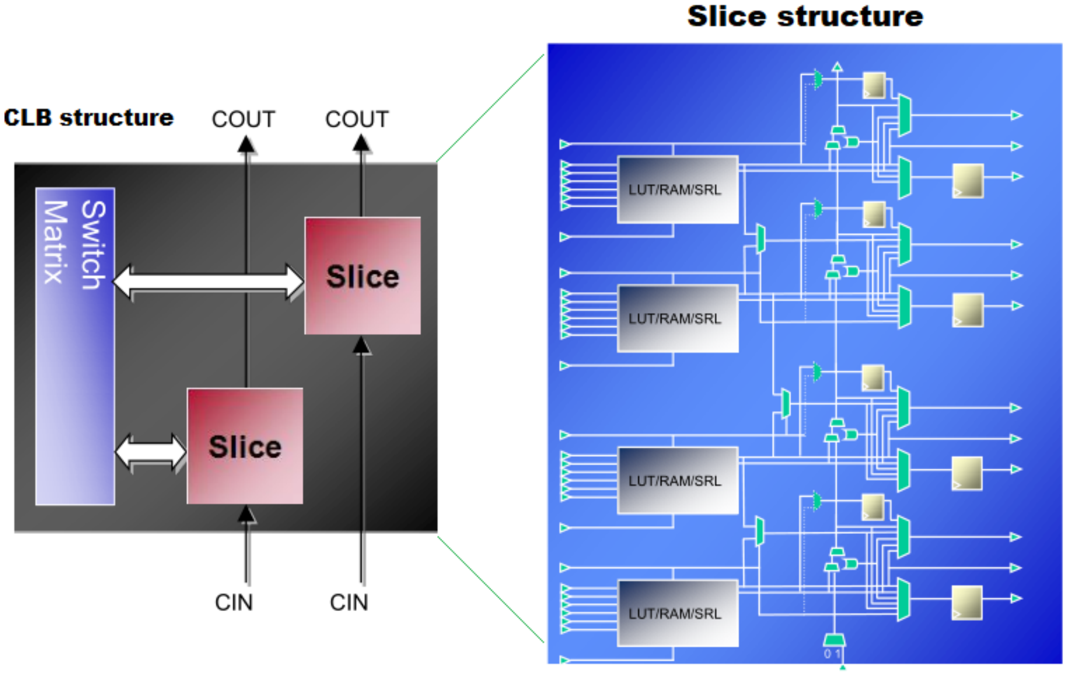


Figure 3.2: FPGA CLB representation. Left panel reports the communication scheme between slices in a CLB through a switch matrix. The zoom in the right panel shows the details of the slice structure, including LUT, MUX, FF resources.

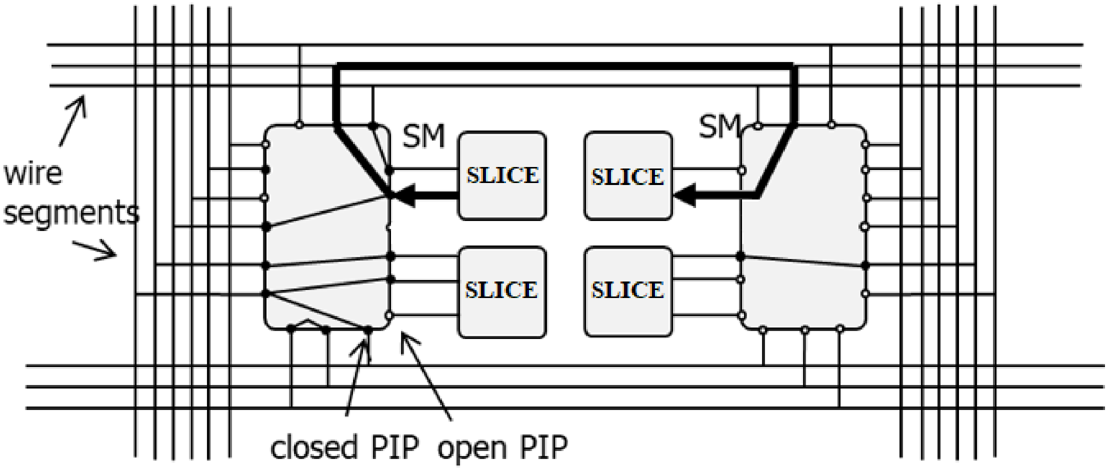


Figure 3.3: Representation of switch matrices making communication through slices.

- **Input/Output blocks (IOBs).** IOBs are in the periphery area and represent the FPGA interface for external connection. These blocks include several components as FFs, resistors, buffers and pads to implement the input and output functions of an FPGA. The FFs are available for data synchronization in order to manage sequential signals. There are buffers to manage different logic standards and to calibrate the input/output current basing on signal fanout. PADs are available to getting data, clocks, power etc, in and out to the FPGA.

- **Clock Management Blocks.** FPGA has dedicated clock management blocks allowing to generate new clock signals and efficiently distribute them in the system.

- **Block RAM (BRAM).** The BRAM is a dual-port RAM module embedded into the FPGA fabric to provide on-chip storage for a relatively large set of data. The two types of BRAM memories available in a device can hold 36k bits, and the available amount of these memories is device specific. Each 36 Kb block RAM can be configured in several ways, e.g. as a 64K x 1, when cascaded with an adjacent 36 Kb block RAM, or 32K x 1, or 16K x 2, etc.

The dual-port nature of these memories makes it possible for parallel, same-clock-cycle access to different locations.

- **Digital Signal Processing (DSP).** The DSP block is an arithmetic logic unit (ALU) embedded into the fabric of the FPGA, composed of a chain of three different blocks. Among the sub-components of a DSP, there are an add/subtract unit and a multiplier.

A typical FPGA, e.g. XC7K325T, includes 203800 LUTs, 152850 FFs, 445 BRAMs, 840 DPSs and 10 IO banks.

A crucial role in FPGAs is played by programmable routing for which interconnections are realized by properly shorting or opening switches between predefined metal lines. Programmable routing is available through programmable interconnection points (PIPs) or through arrays of PIPs, represented by switch matrices. Furthermore, there are different length lines for local and global routing and special networks for clock distribution. These networks are designed to minimize timing skew and jitter.

The design flow to realize a circuit in a FPGA includes four steps (see Fig. 3.4). The first step consists in a description of logic - for example by means of a hardware description language (HDL)- and of requirements through design constraints. The second step is the logic synthesis where a designated tool translates the description to a list of interconnected logic primitives, e.g. LUTs, FFs, DSPs, IOBs, and so on. In the third step, placement and routing of the circuit are executed by placing the mentioned primitives and realizing interconnections via programmable routing. Moreover, static timing analysis and digital simulation at several intermediated steps are performed. The last step is the bitstream generation, which provides the configuration file to be loaded into the FPGA for implementing the design.

The FPGA configuration memory in 7-series devices, is based on SRAM cells and is logically arranged as a matrix, where rows indicate different clock region and each row is an horizontal slice divided in columns. Columns configure represent different available resources and consist of frames, which are and the smallest accessible unit of configuration memory. The number of frames in a column depends on the resources it configures (e.g. CLB, BRAM, DSP, IO etc). In 7-series devices, a frame is 3232 bit long and the column size ranges from 28 to 128 frames (see Fig. 3.5).

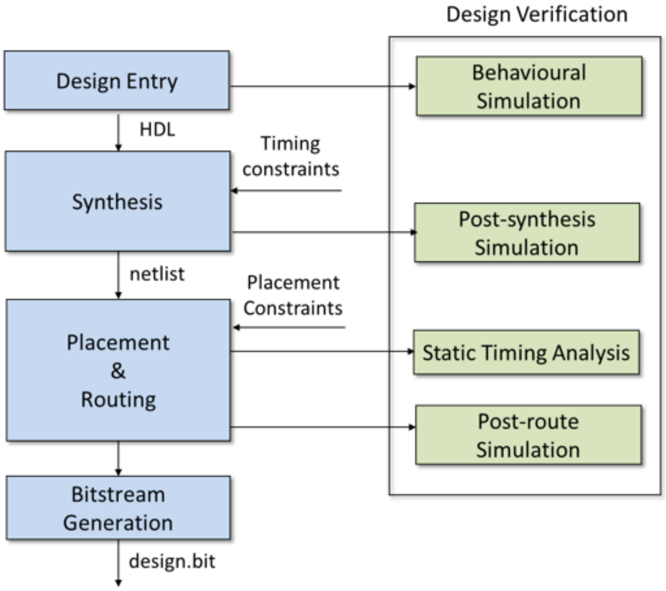


Figure 3.4: Design Flow for FPGA circuit design and implementation

The bitstream is a collection of commands and data, and it can be loaded into the device through several interfaces, e.g. the JTAG port, BPI, SPI, the SelectMAP interface, and so on. When the bitstream is loaded into the FPGA, it generates a sequence of reads and writes to dedicated registers. This makes it possible to supply commands to properly initialize these registers according to the needed behaviour. Such operations are performed before and after loading actual configuration in the frames. For instance, a frame write requires four steps:

- write an identification code (IDCODE) associated to the device, to the IDCODE register;
- write the frame address to the Frame Address Register (FAR);
- write to the Command register to specify the operation on the configuration;
- write to the register dedicated to the frame data, named FDRI.

Furthermore, the access to the configuration can be performed into the fabric trough the Internal Configuration Access Port (ICAP), as shown in Fig 3.6

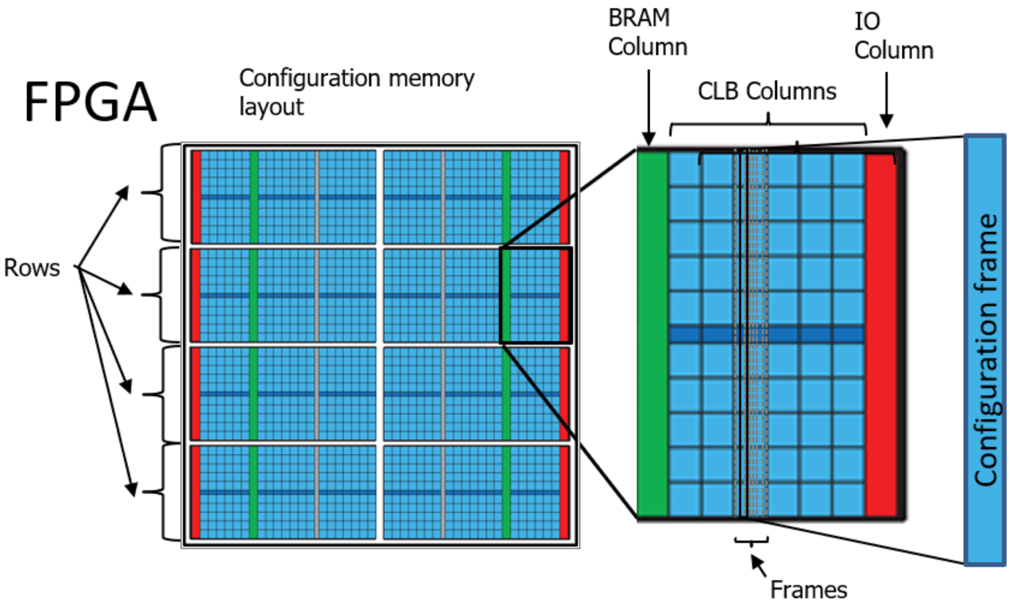


Figure 3.5: FPGA configuration memory architecture. [27]

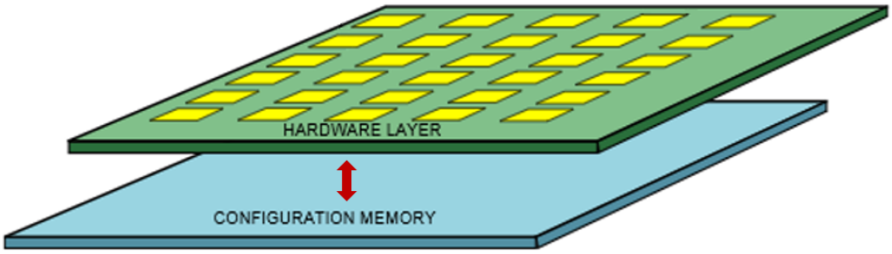


Figure 3.6: Representation of access to the configuration trough ICAP

## 3.2 Soft errors in SRAM FPGAs

Modern SRAM-based FPGAs include up to hundreds Mb of SRAM for hosting the configuration. As a consequence, there are many memory cells sensitive to SEUs, which might alter the design functionality as shown in Fig. 3.7. SEUs in configuration memory are the major concern among the possible SEUs in all the sequential elements of the device. Indeed, when SEUs happen in the configuration memory, their effects last since they remain latched until the configuration memory is rewritten with new configuration data.

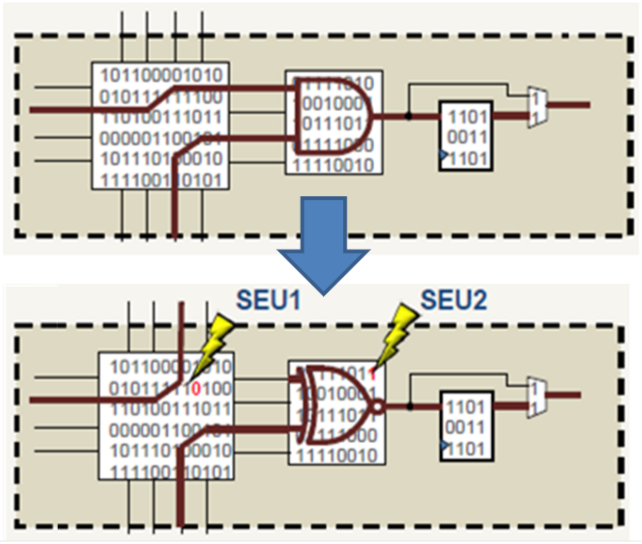


Figure 3.7: Example of SEUs effects in a FPGA configuration memory.

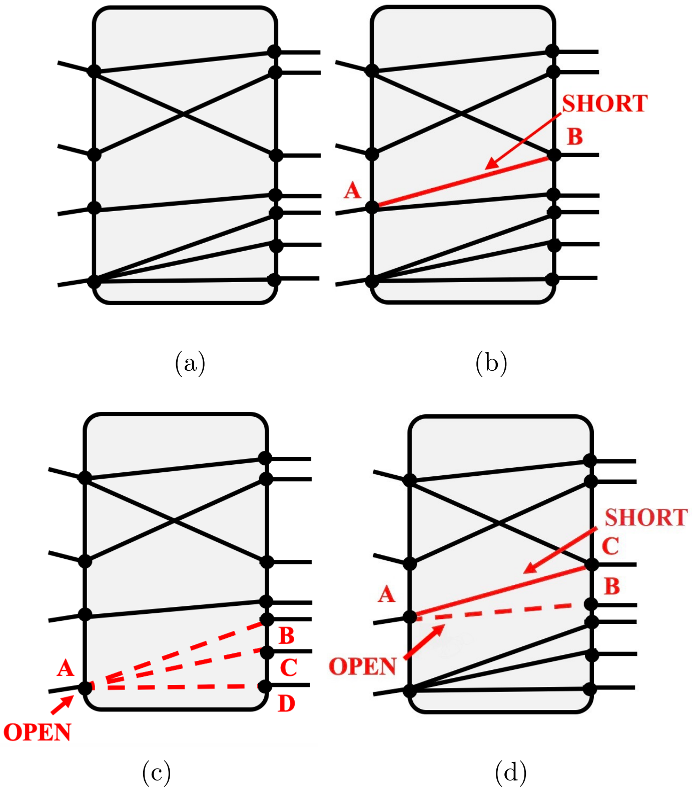
Studying SEU effects on FPGA, it is important to gather information about which resources are actually used by the implemented circuit. Typically, excluding BRAMS, about 90% of the configuration memory bits configure routing resources.

The errors produced by SEUs in the FPGAs configuration memory can be classified into two different categories: logic blocks and routing errors. Different phenomena may be observed in logic blocks errors, depending on resources affected by SEU. For example, the SEU can modify one LUT bit, changing the combinational function implemented, or a FF, altering polarity of the reset line or of the clock line. Moreover, a SEU can modify the configuration of a MUX in the logic block, causing wrong signals forwarding inside the logic block.

As far as SEU in switch matrices is concerned, although it modifies the configuration of one PIP, both single and multiple effects can be originated. Single effects occur when alteration caused by SEU modifies only the logic state of corresponding PIP. In Multiple effects can be explained as modification of connections depicted in Fig. 3.8(a), and can be split in three categories, described below.

- **Short.** SEU modifies PIP A−B, adding a short between them, as depicted in Fig. 3.8(b). This effect can happen when A and B belong to the same switch matrix.

- **Open.** The Open effect occur when a PIP controls more than one connection. This effect corresponds to the deletion of a connection A−B, A−C and A−D and is depicted in Fig.3.8(c).
- **Open-Short.** This effect occur when a SEU influences a PIP controlling at least two connection and is represented in Fig. 3.8(d) as deletion of A−B connection and the creation of A−C connection.



(a) (b)

(c) (d)

Figure 3.8: Possible multiple effects on PIP caused by a SEU. (a) Representation of connection in a switch matrix. (b) Short effect represented by the creation of a bridge between A−B. (c) The open effect causes the deletion of A−B, A−C, and A−D connections. (d) The open-short effect is depicted through the interruption of an existing connection between A and C and the A−B bridge.

Furthermore, experimental data performed on Xilinx Virtex-5 LX50T FPGA with 62-MeV proton irradiation show an increase of the "static current" with SEUs accumulation [28]. The static current $I_s$ is defined as the current absorbed while the device is configured with the design of interest, but it is not clocked. $I_s$ includes two terms: the first term $I_0$ depends on the specific device and the second term $I_{res}$ depends on the specific configuration. Since SEUs alter the FPGA configuration, $I_{res}$ is a function of the number of accumulated SEUs. Then, the static current can be written as

$$I_s = I_{s0} + I_{sres}(n), \tag{3.1}$$

where n is the number of accumulated SEUs. Moreover, it is possible to define the dynamic current $I_d$ as the difference between the total current drawn by the device when it is configured and clocked and the static current.

$$I_d = I_t - I_s. \tag{3.2}$$

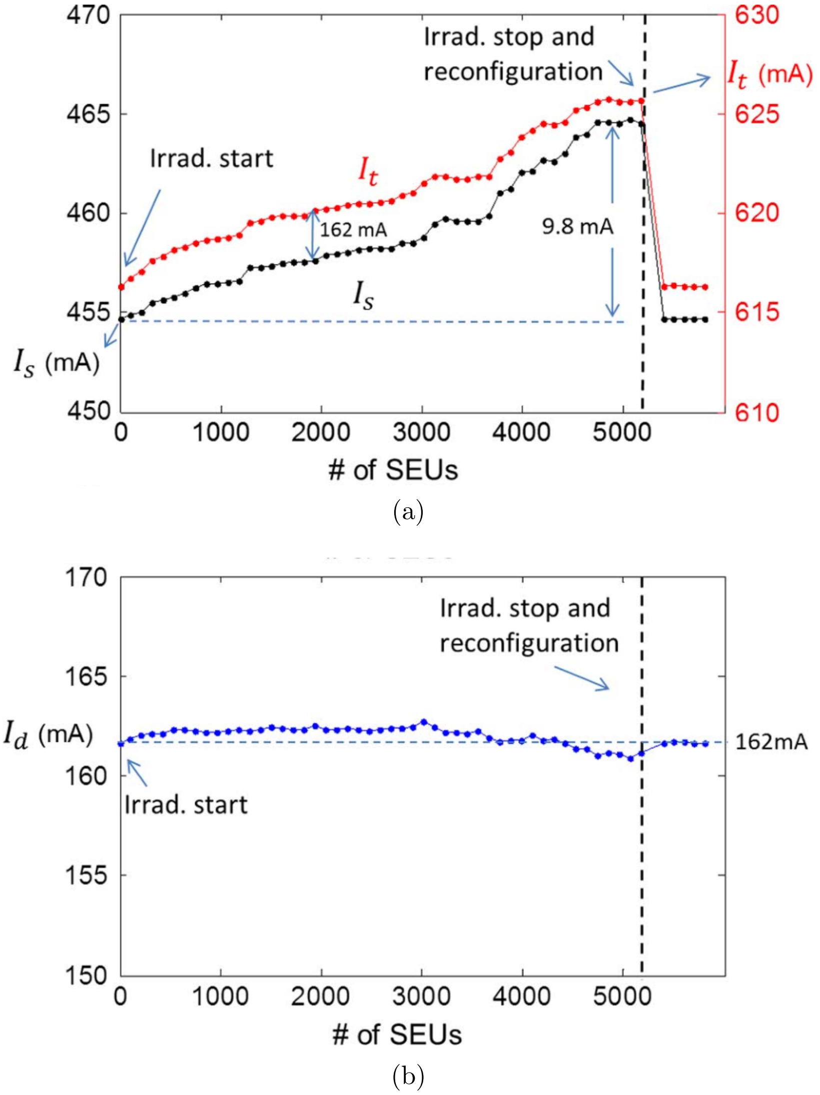
Analogously, the dynamic current can be split into two contributes

$$I_d = I_{d0} + I_{dres}(n), \tag{3.3}$$

The irradiation test results show that the static current increases with the number of accumulated SEUs (see Fig. 3.9(a)), whereas the variation of dynamic current can be neglected as shown in Fig. 3.9(b). The above-mentioned effects show the FPGA sensitivity to radiation-induced upsets in the configuration memory. Thus, in HEP environments, the development of mitigation techniques on SRAM FPGAs is needed to preserve the intended functionality of the circuit.

## 3.3 Mitigation techniques

The solutions able to mitigate SEU effects on FPGAs configuration memory can be split in two main categories: redundancy- and reconfiguration-based methods [29]. Although they share the same aim, such methods operate at different levels. Indeed, the first group masks the propagation of SEUs, which might alter the design functionality effects to the circuit outputs, whereas the second one directly acts on the configuration, restoring the original values into configuration bits after the alterations induced by SEU. The masking does not prevent the accumulation of SEUs, thus giving an excess of power consumption, in addition to a decrease of the circuit reliability, which will be more deeply discussed in 3.3.2. However, redundancy-based



(a)

(b)

Figure 3.9: Test results on current trends for a Virtex-5 LX50T irradiated with a 62-MeV proton fluence of $2.3\times10^{10}$ cm$^{-2}$. (a) Graphic of static current versus accumulated configuration SEUs. (b) Graphic of dynamic current versus accumulated SEUs.

methods are advantageous when combined to the reconfiguration-based ones. In the next subsections, some examples of the two mitigation technique categories will be described, in addition to the combined cases.

### 3.3.1 Configuration scrubbing

Configuration scrubbing is a technique for removing configuration SEUs effects. It can be performed in several ways, including the following.

- **Blind scrubbing.** Blind scrubbing is the periodical overwriting of the configuration retrieved from a "golden" memory, assumed to be error-free. This technique is fast and reliable but does not permit to obtain information about occurred upset and it requires external components.
- **Readback scrubbing.** In this implementation, the scrubber performs a real-time comparison of the readback configuration from the device and the configuration stored in the "golden" memory. Thus, the reconfiguration is accomplished only when differences are detected and the overwriting operation involves only the corrupted frames.

The first method performs a full reconfiguration of the FPGA memory configuration, whereas the second method performs a partial reconfiguration when some differences are detected. Others readback scrubbing techniques can be implemented by means of error correcting codes (ECCs). For instance, the Xilinx Soft Error Mitigation (SEM) controller, which is provided by Xilinx and will be described in Sec. 3.3.4. The scrubber can be internal or external to the device as schematized in Fig. 3.10. An internal scrubber makes the system more compact but can be impacted by SEU which can affect also its functionality. As far as the external-scrubber implementation concerned, it is possible to place it in a radiation-free zone in order to safeguard its functionality by avoiding SEUs effects on its configuration memory.

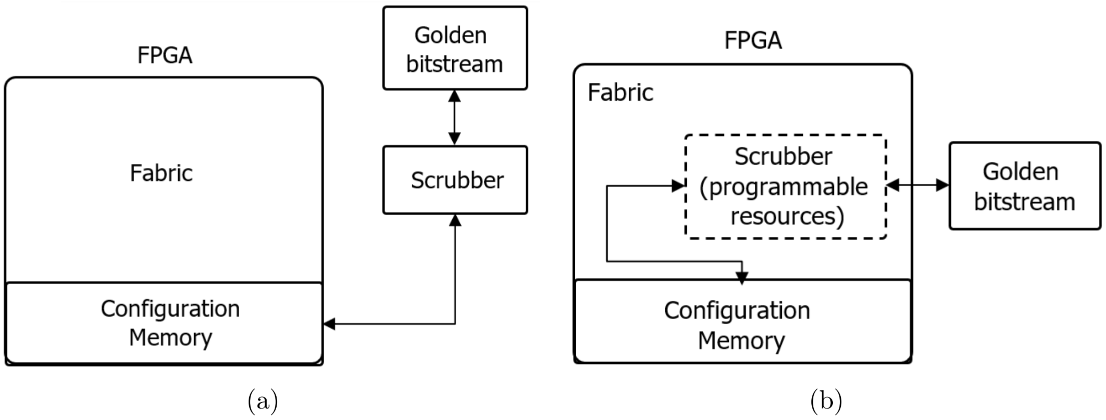


Figure 3.10: Implemented scrubber on FPGA. (a) External implemented scrubber. (b) Internal implemented scrubber.

The readback process is often transparent to the implemented circuit, which continues to operate normally even while the readback process is running. As a consequence, the partial reconfiguration mechanism permits to rewrite the configuration data without interrupting the operation of the circuit [29]. The scrubbing method in the form of partial or full reconfiguration is mandatory for adopting SRAM-based FPGA in the presence of SEUs. Indeed, these techniques are the only viable solution to prevent the accumulation of soft error within the configuration memory.

### 3.3.2 Triple Modular Redundancy

The redundancy-based techniques adopt additional hardware components or additional computation time for detecting the presence of SEUs and/or masking SEUs propagation to the circuits outputs. For instance, faults detection can be achieved through the duplication of the circuit implemented on the FPGA by generating two replicas and comparing their outputs, i.e. by means of the Duplicate With Comparison (DWC) technique (see Fig. 3.11). Thus, when a difference is detected, a signal is produced as soon as a mismatch is found. When fault masking is mandatory, the designer may choose the Triple Modular Redundancy (TMR) approach. TMR is realized by designing three copies of the same circuit and building a majority voter on replicated circuits outputs as shown in Fig. 3.12. This solution is a widely used technique that permits only a masking of SEUs configuration effects, without implementing a correction of configuration. As a consequence, the TMR implementation does not prevent SEUs accumulation and also reduces circuit reliability.

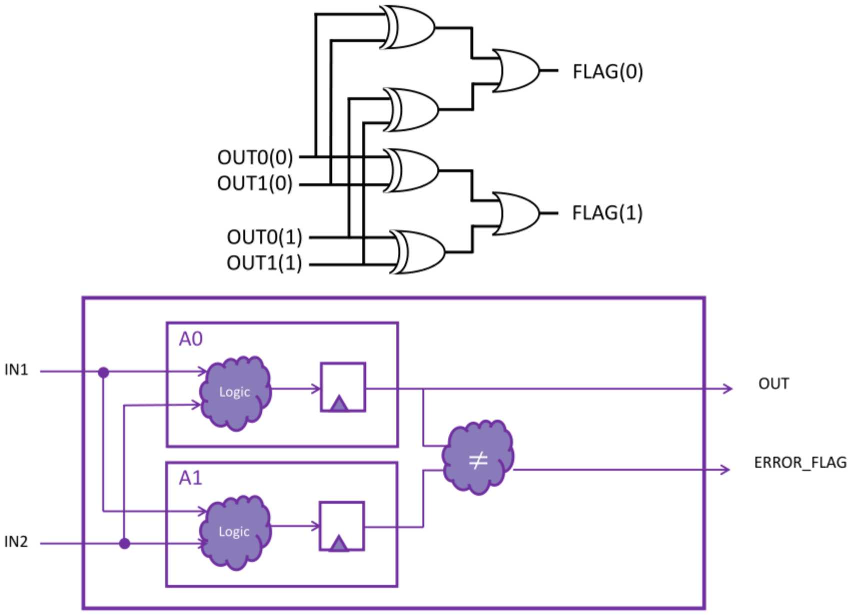


Figure 3.11: Representation of DWC implementation. When a difference in comparison process is detected the flag bus reports an error code.

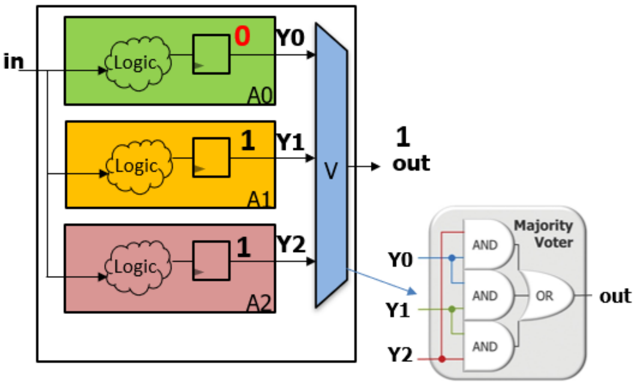


Figure 3.12: Representation of TMR implementation. The majority voter result masks the wrong output value of one circuit copy, preserving the intended original circuit functionality. The majority voter can be obtained as OR of the AND of three output values.

The reliability $R(t)$ is the probability that a system operates correctly at a time $t$. Considering a number N of systems , it is related to the failure rate $\lambda(t)$, i.e. the number of failed systems $(N_f(t))$ in a given time interval $dt$, normalized with respect to the number of survived systems $(N_s(t))$. By definition

$$R(t) = \frac{N_s(t)}{N}, \quad \lambda(t) = \frac{1}{N_s(t)}\frac{dN_f(t)}{dt}. \tag{3.4}$$

The time derivative of $R(t)$ gives

$$\frac{dR(t)}{dt} = -\frac{1}{N}\frac{dN_f(t)}{dt} = -\frac{\lambda(t)N_s(t)}{N} = -\lambda(t)R(t). \tag{3.5}$$

Performing the time integral, the equation becomes

$$\ln\frac{R(t)}{R(0)} = -\int_0^t \lambda(t')dt'. \tag{3.6}$$

Assuming that the system correctly operates at the initial time, i.e. $R(t=0)=1$, the final expression for reliability is

$$R(t) = e^{-\int_0^t \lambda(t')dt'}. \tag{3.7}$$

In a system with a constant failure rate $\lambda(t) = \lambda$, the system reliability $R_s(t)$ decays exponentially:

$$R_s(t) = e^{-\lambda t}, R(t=0) = 1. \tag{3.8}$$

Then, it is possible to estimate the mean time to failure $(MTTF)$

$$MTTF_s = \int_0^\infty R_s(t)dt = \frac{1}{\lambda} \tag{3.9}$$

For a system with implemented TMR, reliability and $MTTF$ are

$$R_{TMR} = 3R^2 - 2R^3 = 3e^{-2\lambda t} - 2e^{-3\lambda t}, \tag{3.10a}$$

$$MTTF_{TMR} = \frac{3}{2\lambda} - \frac{2}{3\lambda} = \frac{5}{6\lambda} < MTTF_s. \tag{3.10b}$$

Therefore, mean time to failure in TMR systems is lower than the MTTF of the single modules, as shown in Fig. 3.13. In order to have a beneficial impact of the TMR on reliability, $R_s(t)$ must be larger than 0.5.

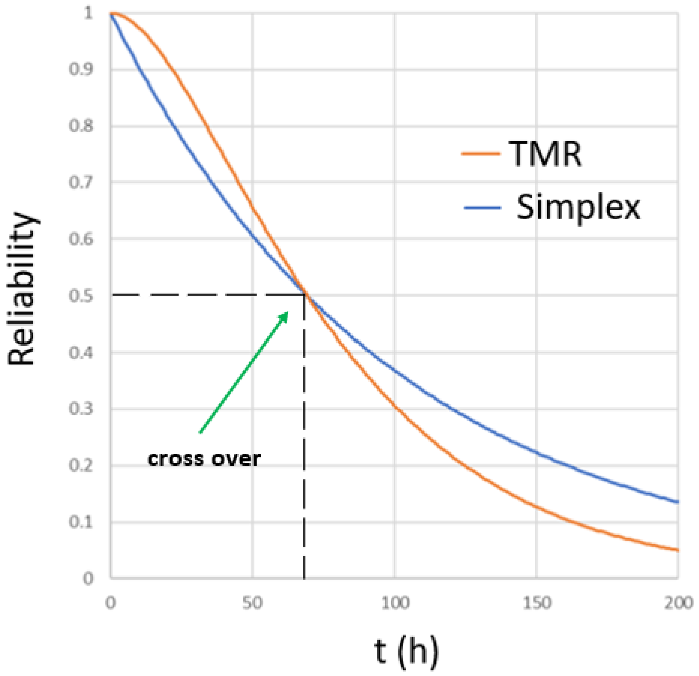


Figure 3.13: Reliability in function of time for a single system and a TMR system. $MTTF_s = 100h$ and the cross over occur for $R(t) = 0.5$ and $\lambda t = 0.693$

So far, TMR has been applied to a single circuit, considered as a unique block. An improvement can be obtained by applying TMR to the various circuit components, i.e by applying the selective TMR (STMR) [29]. Unlike the simple TMR, where the whole circuit outputs are voted, STMR replicates only the portions of a circuit that are more sensitive to SEUs. Indeed, STMR method can have beneficial effects on power consumption, due to the selectively triplication, and on circuit reliability. The disadvantage is the larger number of voters required, since there must be a voter for each component.

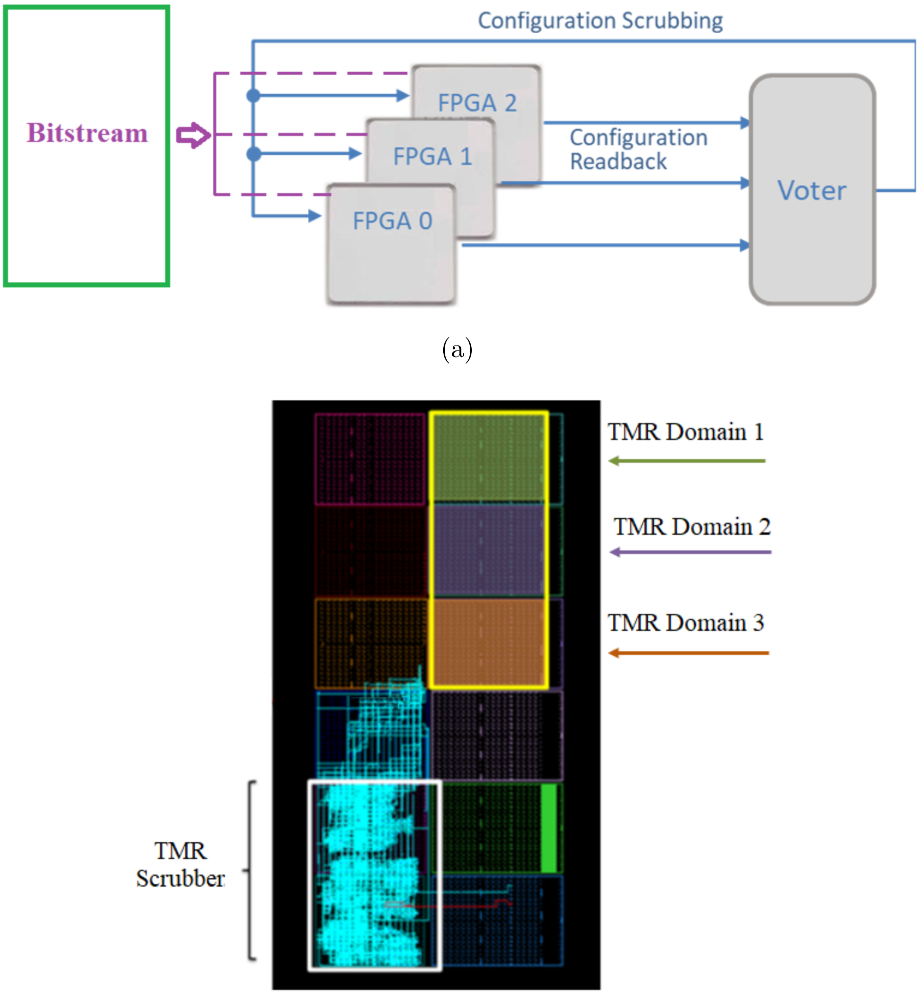
Experimental data show that a single modification in the configuration memory cell dedicated to routing or logic is able to produce multiple errors. Furthermore, experimental data on several benchmark circuits, designed according to the TMR architecture, show that ~14% of the configuration memory bits upset, affecting cells storing information about routing resources, produce multiple errors that the TMR is unable to mask [30]. In particular, an analysis on Xilinx Virtex family FPGAs shows that multiple effects occur when a switch matrix with two implemented PIPs, routes signals pertaining to two different TMR replicas. The mentioned test has been realized by comparing each faulty configuration with the corresponding image of the configuration memory. It shows that 72% of all the configuration memory bits controlling the considered switch matrix could produce critical situations, if used for routing different TMR replicas. Therefore, when TMR is used by the designer, the triplication of routing is mandatory. With this scenario, redundancy-based techniques are not sufficient by themselves to ensure complete reliability against SEUs effects. Thus, the TMR architecture helps in reducing the number of escaped SEUs, but it is unable to completely remove them.

### 3.3.3 TMR-based configuration scrubbing

An interesting solution for improving the reconfiguration-based techniques is implementing a TMR-based scrubber. In this scrubber implementation, three circuits with identical bitstream and operating on the same inputs are realized. Scrubbing is performed as partial reconfiguration, i.e. the memory configurations of the three circuits implemented, are written only when the TMR voter result differs from read-back values. The TMR-based scrubbing can be realized as externally or internally to the FPGA.

- **TMR-based external scrubbing.** This method is implemented by using redundant FPGAs configured with identical bitstreams and is shown in Fig. 3.14(a). The TMR-based external scrubbing implementation is simple and has a very high correction capability and a low repair latency. Moreover, TMR permits SEUs mitigation effects on circuit functionality. On the other hand, this method needs a radiation-hardened voter and three FPGAs are required to generate redundant configuration with consequent triplication of power and cost.

- **TMR-based internal scrubbing.** In TMR-based internal scrubbing, depicted in Fig. 3.14(b), the redundant modules are realized in the same FPGA. In addition to the benefits of the previous solution, i.e. high correction capability and low repair latency, this approach does not require redundant FPGAs. Furthermore, this implementation realizes a TMR on both circuit functionality and configuration. On the other hand, the redundant modules triple the power consumption by tripling the current consumption related to resources $(I_s(n) + I_d(n))$, and require a strategy for generating identical configurations for all modules in the device. Moreover, an internal scrubber can not be protected by SEUs which can alter its functionality causing a decrease of the system reliability.



(a)

(b)

Figure 3.14: TMR-based scrubbing implementations. (a) TMR-based external scrubbing implementation. (b) TMR-based internal scrubbing implementation on FPGA configuration memory.

The described solutions allow to correct configuration without interrupting circuit operations and to avoid the usage of radiation-hardened off-chip memories as “golden” image, exploiting the SEUs effects mitigation of TMR.

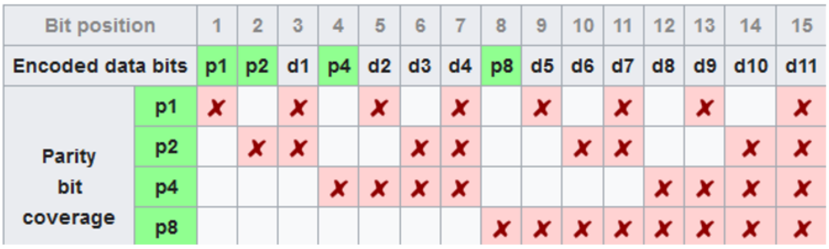
### 3.3.4 SEM controller

7-series Xilinx FPGAs include the Readback Cyclic Redundancy Check (CRC) that is a hardware performing continuous readback of configuration frames [31]. In particular, the Readback CRC calibrates ECC bits and CRC value when the bitstream is loaded into the FPGA. The calibration of the ECC bits is performed by using the Hamming code, whereas the CRC value is calculated using CRC code. Furthermore, as previously mentioned, 7-series devices support the SEM IP scrubber, which is designed for very low upset rate environments, e.g. the earth atmosphere, and is not radiation-tolerant. The SEM performs SEU detection, correction, and classification for configuration memory, by clocking and observing - through the ICAP and FRAME_ECC dedicate Xilinx primitives - the Readback CRC hardware as part of the SEU detection function.

Hamming code and CRC code are widely used and are represented in Fig 3.15. Hamming code is a linear parity-based code, performing single error correction and double error detection (SECDED). Indeed, it can detect up to two-bit errors or correct one-bit errors without detection operation. The Hamming code is executed by adding to the original message a series of redundant bits, named as parity bits. Parity bits can be calculated according to:

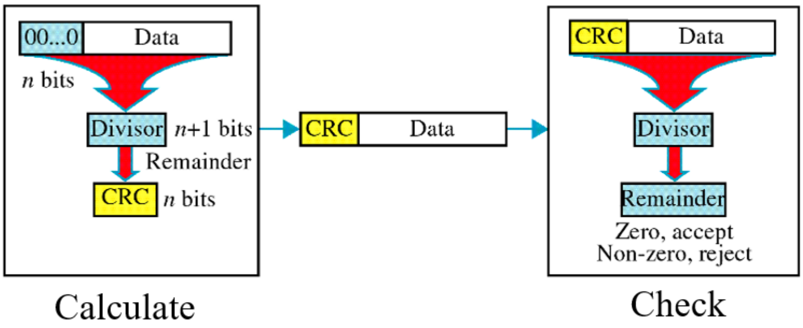
- even parity, where the parity bits is set to make the total number of set bits even;
- odd parity, where the parity bits is set to make the total number of set bits odd.

Once the parity of Hamming code implementation is defined, each parity bit $P_i$ is calculated on a subset of the data word. Given the data word, the parity bits are inserted in the bit positions that are powers of two, and the data bits in the remaining positions, as shown in the encoded data bits line in Fig. 3.15(a). The specific positioning of extra bits makes it possible to detect and correct single-bit errors performing recalculations and check of parity bits which permits to determine the error position. For $m$ data bits, $n$ parity bits are required, with $2^n \geq m+n+1$, exhibiting a logarithmic growth of parity bits. For instance, 11 data bits require 4 parity bits whereas 1000 data bits require 10 data bits. The choice of the parity, even or odd, is irrelevant but the same choice must be used for both encoding and decoding. The code word is obtained by replacing the $n-$th parity bit by the XOR result of the bits at the positions having the $n-$th least significant bit set.



| Bit position | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Encoded data bits | | p1 | p2 | d1 | p4 | d2 | d3 | d4 | p8 | d5 | d6 | d7 | d8 | d9 | d10 | d11 |
| Parity bit coverage | p1 | ✗ | | ✗ | | ✗ | | ✗ | | ✗ | | ✗ | | ✗ | | ✗ |
| | p2 | | ✗ | ✗ | | | ✗ | ✗ | | | ✗ | ✗ | | | ✗ | ✗ |
| | p4 | | | | ✗ | ✗ | ✗ | ✗ | | | | | ✗ | ✗ | ✗ | ✗ |
| | p8 | | | | | | | | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |

(a)



(b)

Figure 3.15: The widely used ECC code: (a) Hamming code bit parity representation. (c) CRC calibration and check description.

CRC codes add a fixed-length check value to the provided data, in order to verify the message correctness. Cyclic codes are particularly well suited for the detection of burst errors, i.e. contiguous sequences of erroneous data symbols in a messages. The specification of a CRC code requires definition of a polynomial generator, which represent the divisor in a polynomial division. We define a $n-$bit CRC a code generating a $n-$bit long check value. For a given $n$, multiple CRCs are possible, each with a different polynomial. Such a polynomial has highest degree $n$, i.e., it has $n+1$ terms. The simplest error-detection system is the 1-bit CRC, where the added bit represents the parity bit. CRC is based on the principle that it is very unlikely to get the correct remainder if the data gets altered by random events.

In 7-series FPGAs, the Hamming code is used for detection at frame level by exploiting dedicated bits included in the frame, for tracking an erroneous behaviour and correcting it. Instead, the CRC is used for error detection at device level, by calculating the remainder of polynomial division between data and a predefined polynomial. In 7-series devices, the Readback CRC must be enabled by the user through writing to the dedicated configuration register. After that, the Readback CRC achieves continuous readback of configuration data in the background of a user design. In the first round of readback, the ECC syndrome bits are calculated. In the second round, the CRC value is latched as the golden value for the later comparison, or a known value can be supplied setting a specific constraint. At each subsequent readback round, the CRC is calculated and compared to the golden value. In addition to Readback CRC, also correction functionality must be enabled from the user. Moreover, when the latter functionality is enabled, the user can stop or continue the configuration readback after the correction operation. Due to the hardware structure of the Readback CRC, the SEM operation is limited to the correction of a single bit error or to a double bit error, as long as the two flipped bits do not belong to the same frame. Moreover, the SEM can be used for SEUs emulation by injecting errors into the configuration memory. The latter feature is useful in order to evaluate and test the SEU vulnerability of the design without performing irradiation test. Moreover, for SEU classification, the IP core uses Xilinx Essential Bits to further increase system reliability, where essential bits are defined as those bits that are actually used to define the functionality [31, 32].

For many HEP applications, the radiation environments are too harsh for an effective usage of the SEM. Although Xilinx produces FPGAs with radiation-hardened configuration, they are designed for space applications and are too expensive devices (order of 50k$ per unit) for scientific experiments, which might require thousands of units. Therefore, custom solutions for commercial off-the-shelf must be investigated.

In the next chapter a reconfiguration-based technique will be presented, as the main topic of this thesis work.

# Chapter 4

# The Configuration Consistency Corrector in BEAST II

In this chapter, the monitoring system at the Belle II experiment, mentioned in Sec. 1.4.2, is discussed more in depth. In particular, the focus is on the architecture of the developed scrubber circuit and on its two different implementations.

## 4.1 FPGA monitoring system at Belle II

As previously mentioned, BEAST II includes an FPGA monitoring system depicted in Fig 4.1. It is composed of three slices, each including a test board designed by INFN and hosting a Xilinx FPGA, a custom single board computer (SBC)[33], an ADC to measure the voltage at the device, an USB-controlled power supply with three channels for FPGA supply domains, and dedicated cabling. For each slice, the used FPGAs are Virtex-5 LX50T, Kintex-7 70T and Kintex-7 325T, and these are the only active components on the board, in order to ensure that measured failures are unambiguously due to the FPGA. Each SBC is interfaced to the pertaining FPGA for configuring the device via JTAG port and receiving upset details from the implemented scrubber via UART-over-JTAG. Power and configuration are fed to the boards over dedicated cabling from a remote control room at a distance of almost 40 m. The 7-series FPGAs are programmed with a scrubber circuit in order to self-read their configuration and detect upsets, whereas the Virtex-5 one is read back by the SBC, by means of a Tool Command Language (TCL) script.

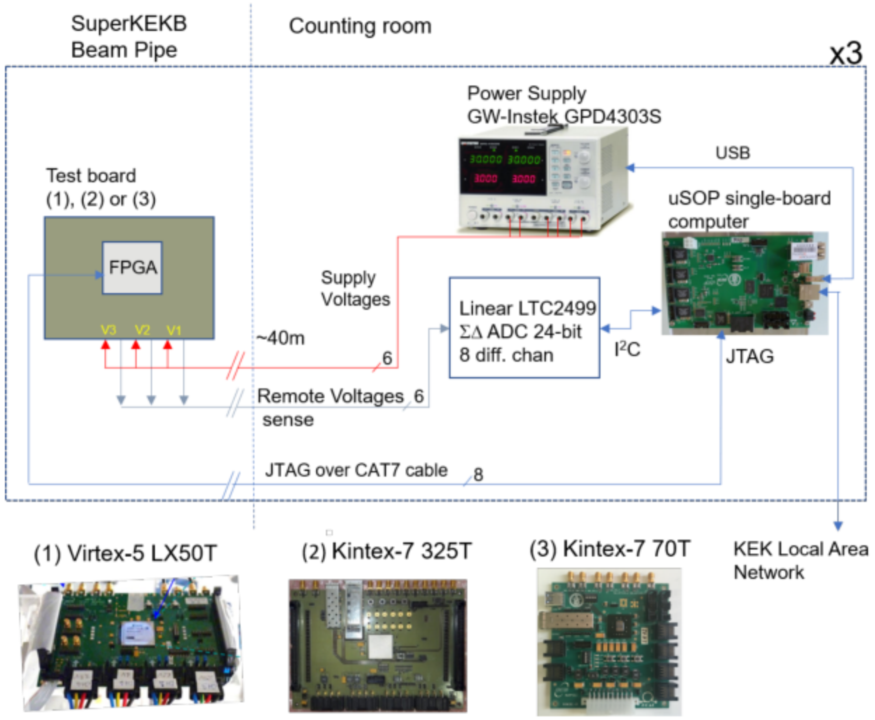


Figure 4.1: Schematic diagram of one slice of FPGAs monitoring system in BEAST II.

The three boards hosting the FPGAs are not equipped with an oscillator and there is no system for clock distribution in BEAST II. For this reason, each FPGA generates an internal 100 MHz clock by means of a digitally-controlled oscillator (DCO) configured in the fabric. Moreover, each scrubber accesses to the configuration via ICAP, continuously scanning frames of its FPGA. The correction operation is performed as partial reconfiguration, and the implemented circuit provides details about detected upsets, including a time stamp, frame address, bit offset, and upset polarity('0'-> '1' or '1'-> '0'). During SuperKEKB operation, information about the power consumption at the different power rails are logged, in addition to details about the upsets, which make it possible to build trends as shown in Fig. 4.2. For each FPGA the upset trend is obtained by considering the number of detected upsets, normalized to the total number of configuration bits of each device. The measurements obtained from April to August 2018 provide two pieces of information:

- the upset trend differs of almost one order of magnitude between Kintex-7 FPGAs at forward (7k325t) and backward (7k70t) spots, related to SuperKEKB collider asymmetry;
- the lower sensitivity of Kintex-7 325T (7k325t) compared to the Virtex-5 LX50T (5v50t) one, shows that the upset rate per bit in newer families has been reduced, despite the technological scaling which works in the opposite direction.

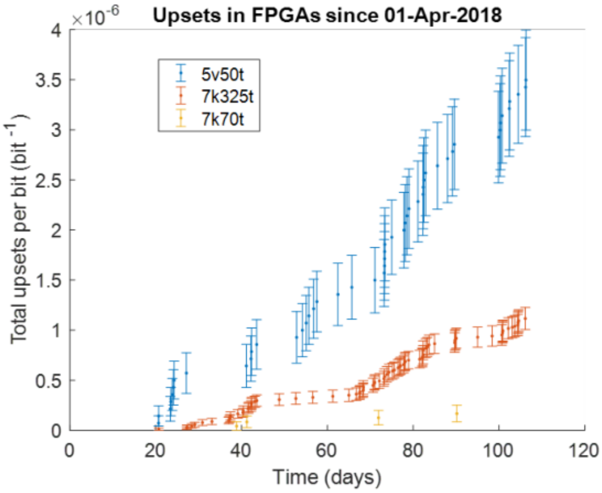


Figure 4.2: Trend of upsets for the three monitoring system FPGAs from April to August 2018.

Moreover, the variation of SuperKEKB current for both the $e^+$ and $e^-$ rings during its operation permits to perform measurements in different radiation conditions.

This thesis work has focused on the development of a scrubber system for the XC7325T FPGA placed on a INFN board represented in Fig. 4.3. Such board is equipped with the JTAG port, a small form factor pluggable (SFP) for optical I/O, a four-wire remote voltage sensing scheme to ensure that the FPGA receives the required voltage, and general purpose input-output (GPIO) pins.

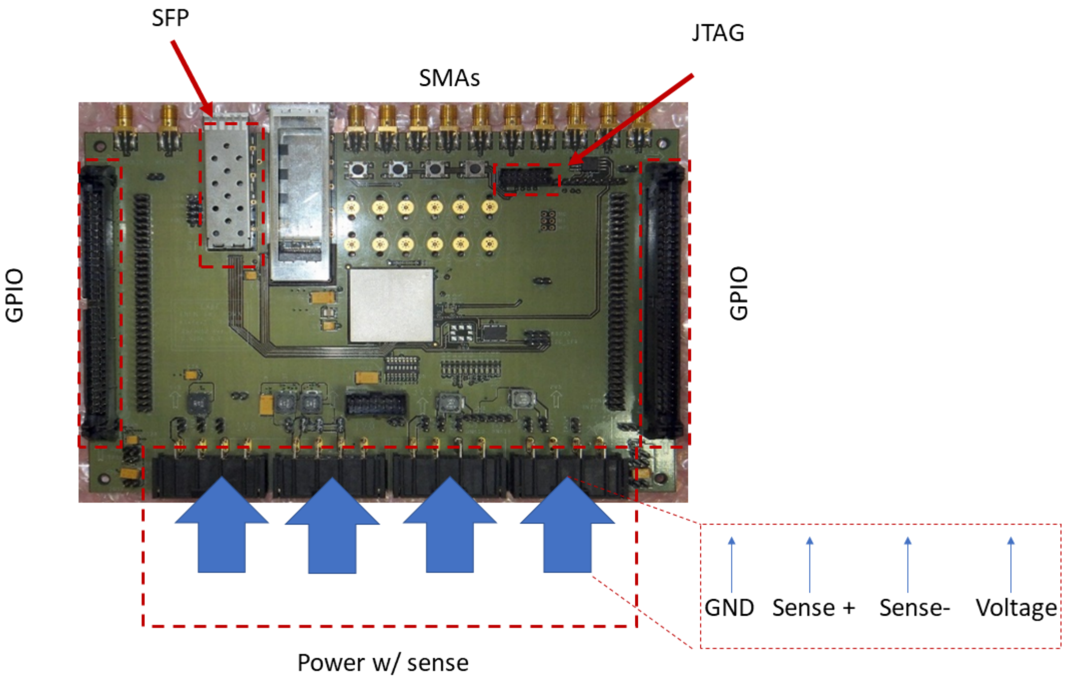


Figure 4.3: Representation BEAST II board hosting the XC7325T FPGA.

In the next sections, the scrubber architecture and its implementations will be in-depth presented.

## 4.2 Redundant-configuration-based Scrubber architecture

In the proposed system for SEUs mitigation, the developed self-repair circuit uses the redundancy of configuration frames to restore the correction configuration, i.e. the proposed circuit is a Configuration Consistency Corrector (C$^3$) scrubber implemented as a triple modular system [27]. This approach requires the identification of used and non-used configuration frames in order to perform the redundancy of the configuration through the following steps:

1. the identification of used and non-used configuration frames;

2. the replication of configured frames to obtain three redundant configuration copies;

3. the identification of the remaining empty frames.

During the FPGA operation, the C$^3$ performs a continuous majority voting of replicated frames (step 2) for error detection and correction. Moreover, empty frames are checked (step 3) to remain in their default status (all bits are zero)[1] and if not they are forced to it. The realized scrubber is implemented in software running on Picoblaze (PB), which is a lightweight 8-bit soft microcontroller provided by Xilinx and supported by most device family. The C$^3$ architecture consists of three cores (Core0, Core1, and Core2), which in turn include the mentioned PB, 4k 18-bit words, a 4k 8-bit words memories, and glue logic for IO (IO ports), as illustrated in Fig. 4.4.

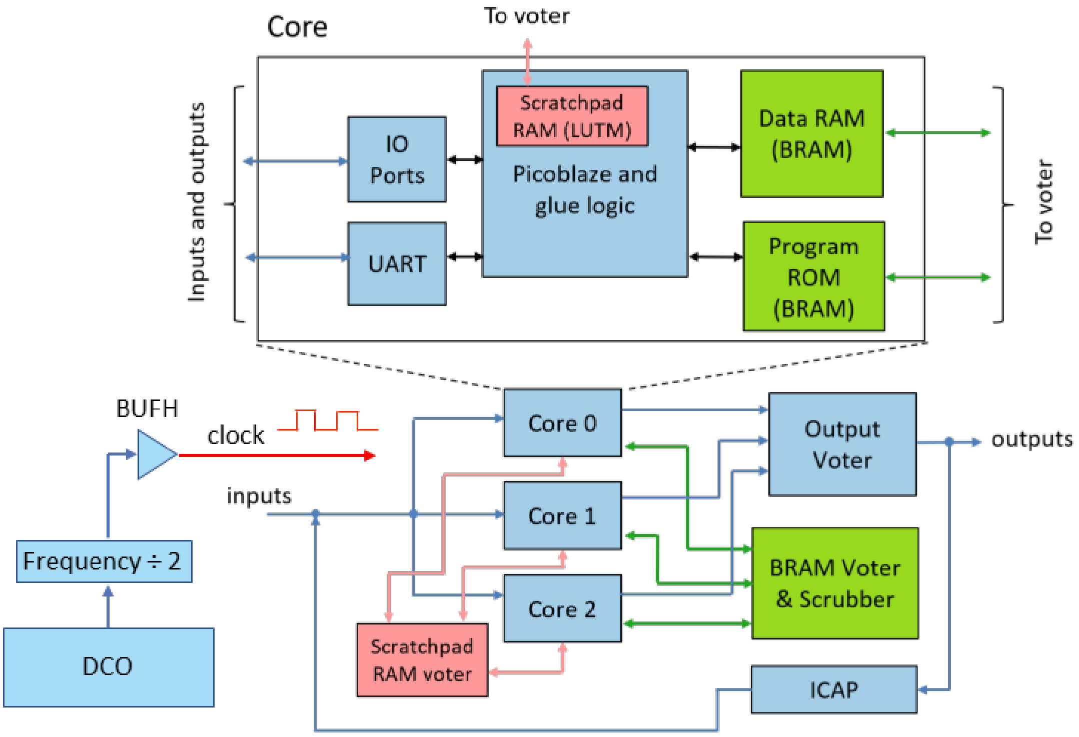


Figure 4.4: Simplified diagram of the proposed scrubber system.

The 4k 8-bit words memory (Data RAM) is used for storing the bits of three copy frames to be voted, the voting result, and the list of frames to be scrubbed.

[1] Although the alteration of empty frames does not influence the circuit functionality, the FPGA static current increases with the number of accumulated SEUs, as shown in Sec. 3.2

Instead, the 4k 18-bit word memory hosts the PB program (Program ROM). The BRAM designated for Data RAM and Program ROM are dual-port BRAMs: one port is accessed by PB and the other one is used by an hardware majority voter for the three cores BRAMs contents. Also, to enhance the $C^3$ reliability, the outputs of IO ports and the core resets are majority voted, and each PB is periodically reset in order to clear possible corruptions of its registered elements (e.g. flags, stack, program counter). At each reset, the PB internal scratchpad memory (128-byte distributed RAM) is majority voted across the three core by a dedicated hardware and a software routine. Finally, BRAMs signals bound to input ports are routed redundantly to each core and all hardware majority voters are in turn tripled and majority-voted.

Other core components are the Data RAM and Program ROM scrubbers, which manage the rewriting of BRAMs contents when a mismatch in one of the cores memories is found.

Regarding the configuration scrubbing, it is performed through majority voters mechanism as well and by preserving the state of the unprogrammed frames. As previously mentioned, the access to the configuration is performed via ICAP, that cannot be tripled due to the FPGA structure. Therefore, a dedicated majority voter for the cores signals destined to the ICAP has been designed, in order to preserve the operation to the FPGA configuration.

Furthermore, the $C^3$ uses the JTAG port for I/O communication and employs the JTAG Loader resource to update the Program ROM at run time. Finally, the DCO generates the clock signal to distribute within the self-repair circuit and there is a counter, named Unixtime counter, which is clocked on the DCO. The JTAG Loader, DCO, and the software of developed scrubber are part of this thesis work as the starting point for the final novel implementations.

### 4.2.1 The digitally-controlled oscillator

The DCO is an oscillator obtained from a specific implementation of digitally-controlled delay lines (DCDLs), which represent a widely used components in timing distribution or trigger and data acquisition systems (TDAQ) of HEP. This system consists in an open-loop fine programmable phase delay used for distributed clocks and/or data lines [34] or to delay signals coming from the detector [35]. The DCDL architecture presented in this thesis work, was conceived at the "Federico II" University by research group of Physics Department and INFN. A simplified diagram of the architecture is shown in Fig. 4.5 [36, 37]. The presented DCDL consists of a multi-input chain of delay elements, a high-fan out distribution network (HFN), and a control encoder. The HFN is a network which distributes a single input signal to multiple delay elements with minimum timing skew. Each delay element implements a logical function of at least three Boolean variables $Y = F(A, B, S_{m-1}....S_0)$, with $m \geq 1$. In the presented system, each chain elements has two inputs: the former is connected to the output of the previous element, whereas the latter is driven by the HFN. For instance, in the DCDL architecture shown in Fig. 4.5, $A$ and $B$ are first and second input, respectively.

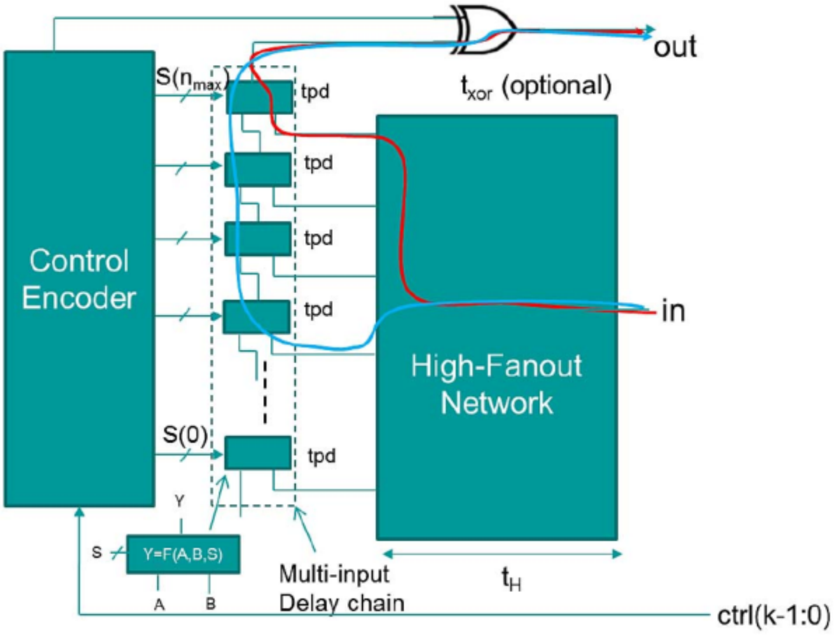


Figure 4.5: Representation of the DCDL architecture. The red path indicates a delay time $t = t_0 + t_{pd}$, whereas the blue path corresponds to $t = t_0 + 4t_{pd}$, i.e. $n = 1$ and $n = 4$ respectively.

The control encoder system manages the signal propagation in the chain: each element can propagate the signal from the previous element or the signal from the HFN in the chain, basing on the input value produced by the control encoder. For example, the $S(0)$ input value manages the propagation of $A$ or $B$ inputs in the first chain element. Furthermore, if needed, each element can invert the signal and a dedicated gate may restore the original signal polarity at the output of the chain. The control encoder sets the $S(i)$ inputs, with $0 \leq i \leq m-1$, by means of a control code $[ctrl(k-1:0))]$, in order to define the number $n = ctrl(k-1:0)$ of elements crossed by the input signal before reaching the delay line output.

Assuming that the delay through each element $t_{pd}$ is uniform and the delay $t_H$ measured between the DCDL input "in" and each chain element input is uniform, the overall delay between "in" and "out" is

$$t = t_H + nt_{pd} + t_{XOR} = t_0 + nt_{pd}. \tag{4.1}$$

Therefore, the delay range is $[t_0 + t_{pd}, t_0 + n_{max}t_{pd}]$, i.e. the minimum chain delay is obtained for $n = 1$, and the maximum delay is obtained for $n = n_{max}$.

The used DCO is obtained by routing the output of the DCDL back at its input through an inverter, as depicted in Fig. 4.6. In particular, the implementation of this system on Kintex-7 325T will be presented. The delay elements are implemented by means of the arithmetic carry propagation primitive of the device (CARRY4), which is the fastest available primitive. Each CARRY4 provides four fast multiplexers (MUXCY) connected in series. A thermometric control encoder has been implemented for driving the CARRY4s selection inputs, and the HFN has been implemented by instantiating a clock buffer in order to access one of the clock trees of the device. The DCO period is twice the loop delay, which takes into account the feedback delay and the DCDL delay:

$$\Delta t_{loop} = \Delta_{feedback} + \Delta t_{DCDL} \tag{4.2}$$

Therefore, the frequency of the implemented oscillator depends on parameters specified in Eq. 4.1, where $t_0$ is related to the system placement with respect to the chosen device clock tree, and the desired $n_{max}$ gives a constraint on the number of carry elements needed. However, the delay chain is obtained by selecting the chain elements crossed by the input signal. In particular, it is possible to split the delay to be achieved in a coarse and fine part, through the $k$ and $k_{dither}$ parameters. Indeed,

$k_{dither}$ makes it possible to achieve a finer choice by means of the dithering approach, i.e. emulating a continuous variation starting from a discrete values scale.

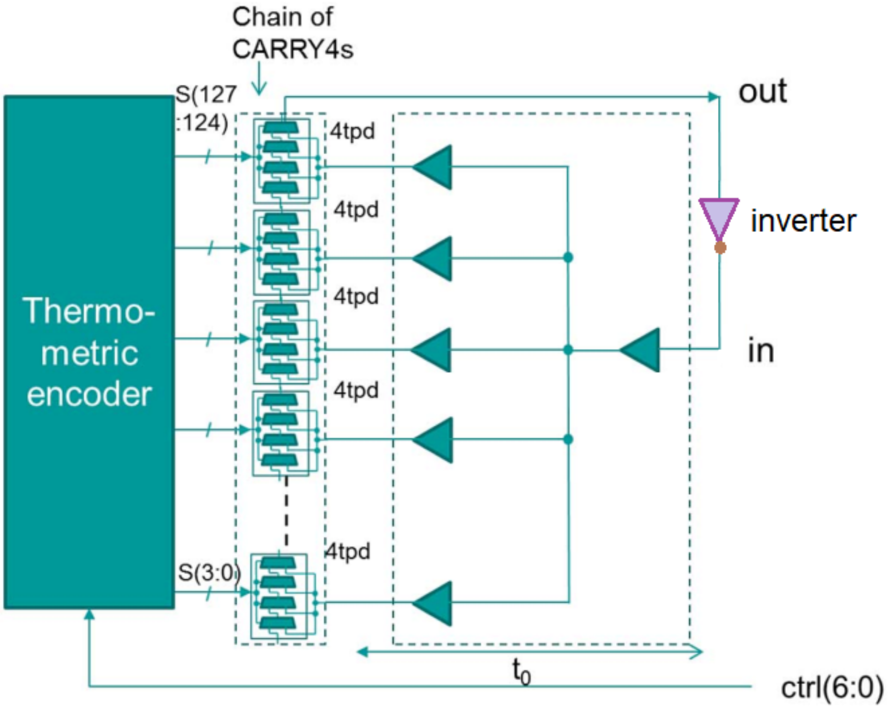


Figure 4.6: Schematic of the DCO architecture.

The implementation of the DCO in a Xilinx 7-series FPGA shows that the DCDL delay depends on the supply voltage $V_{core}$ and the temperature $T$. In particular, test results demonstrate that for $V_{core} = 1V$ and $T \in [40°, 62°]$ range, the delay is well-approximately by an increasing linear function of the temperature, as represented in Fig. 4.7.

Conversely, at $T = 40°$ and $V_{core} \in [0.92V, 1.09V]$ range (entire voltage specification range for the device), the best fit results show that the average loop delay decreases by 4.1 ps/mV, as represented in Fig. 4.8.

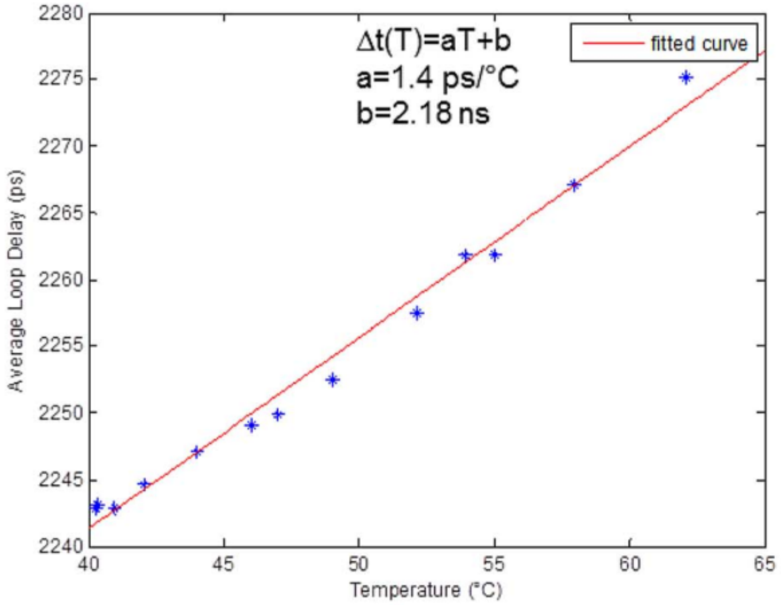


Figure 4.7: Test results showing delay trend of DCO versus temperature $T$ in a Xilinx 7-series FPGA.

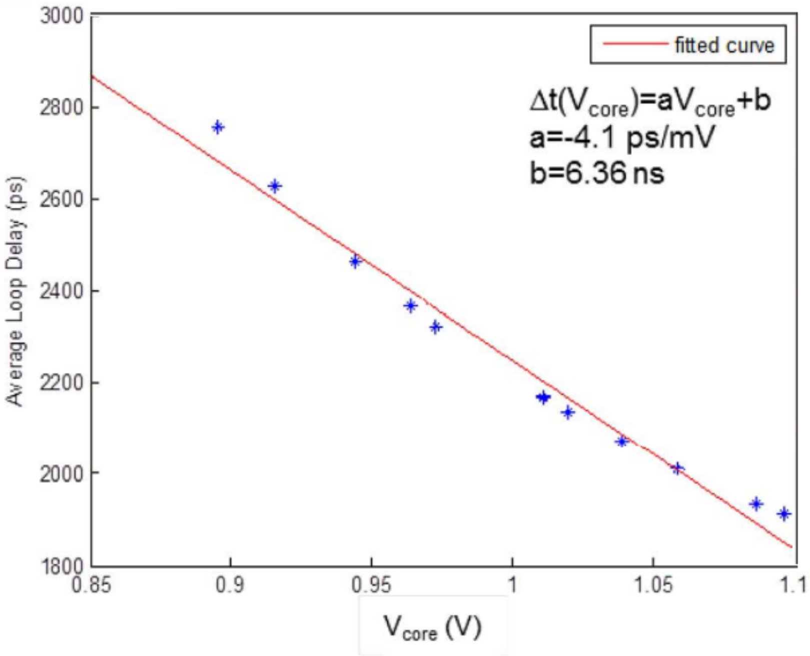


Figure 4.8: Test results showing delay trend of DCO versus the core supply ($V_{core}$) in a Xilinx 7-series FPGA.

Since the DCO frequency is a key parameter, I have worked to find the best possible setting for two implementations. Firstly, the complexity of the circuit and the device limit performances do not allow to use a frequency larger than 100 MHz. Furthermore, in order to reduce the chance that upsets impair the circuit functionality, configuration errors must be removed by using the highest possible frequency and, thus, the ICAP has been used at its maximum specification.

Such a value must be reached to minimize the persistence time of an error within the circuit. Indeed, the larger the persistence time, the smaller the probability to mask the error by using the TMR and, therefore, to correct it.

To set the system frequency, I have designed a DCO calibration system, that is depicted in Fig. 4.9 and described below.

- **200 MHz external clock.** I have used a 200 MHz external clock provided on a Xilinx KC705 off-the shelf evaluation board hosting the Kintex-7 325T FPGA .

- **32 bit counters.** I have realized two counters, named as **EXT-counter** and **DCO-counter**, which count on the 200-MHz and on DCO clock, respectively. The **EXT-counter** is used as comparison term for the DCO frequency.

- **Virtual Input/Output (VIO).** VIO is a Xilinx customizable IP core, that can both monitor and drive internal FPGA signals in real time, and the size of the input and output ports are customizable to properly interface to the design. It has been used to read the counters value, to reset them, and to find the $k$ and $k_{dither}$ value, and the buffer-DCO distance required to obtain an approximation of the 200 MHz frequency.

Since the DCO frequency depends on the temperature and the voltage supply, the Unixtime counter is periodically resynchronized to a PC time with a 5 minutes period, so that the DCO frequency variations on the long run are negligible.

Once I found the parameters for the required frequency, I removed the calibration system, since it requires additional resources which should be also tripled in the redundancy configuration generation (see Sec.4.2 step 2). Moreover, the requirements of the second implementation collide with the DCO calibration system for reasons that will be explained later.

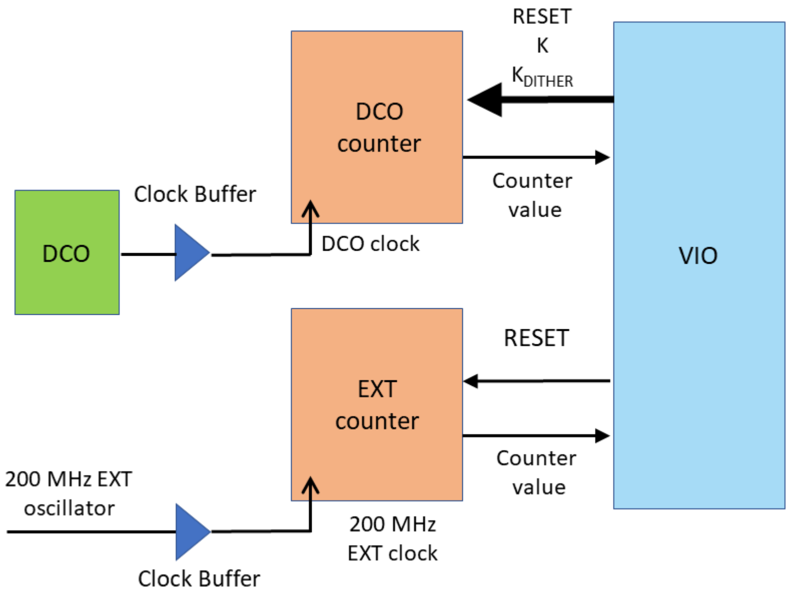


Figure 4.9: Schematic of DCO calibration system.

In the first circuit implementation, I have modified the previous layout in order to optimize the distance between the delay chain and the clock buffer. Then, I have determined the best $k$ for the implemented circuit. In the second implementation, the circuit constraints bounded the DCO position, so that I have changed the $n$ value to reach the required frequency. Furthermore, in addition to the choice of the optimal $k$ value, I have set also the $k_{dither}$ value to reach the desired frequency.

When the 200 MHz frequency has been reached, a frequency divisor generates the 100 MHz signal to be distributed to the system through the selected clock buffer.

### 4.2.2 JTAG Loader

The JTAG Loader, provided by Xilinx as part of the PB design, is an extremely useful tool for software debugging. It gives a way to upload a new program into the BRAM (depicted in Fig 4.10(a)) during the FPGA operation, facilitating the development of PSM code. The JTAG Loader leverages the Xilinx BSCANE2 primitive shown in Fig. 4.10(b), which gives access to the JTAG signals (TDO, TDI, TCK, TMS) in the FPGA. TDO represents the BSCANE2 data input, whereas TDI represents the BSCANE2 data outputs. Although it is possible to upload the BRAM content of several PBs, for the sake of simplicity we will show the JTAG Loader functionality for a single PB.

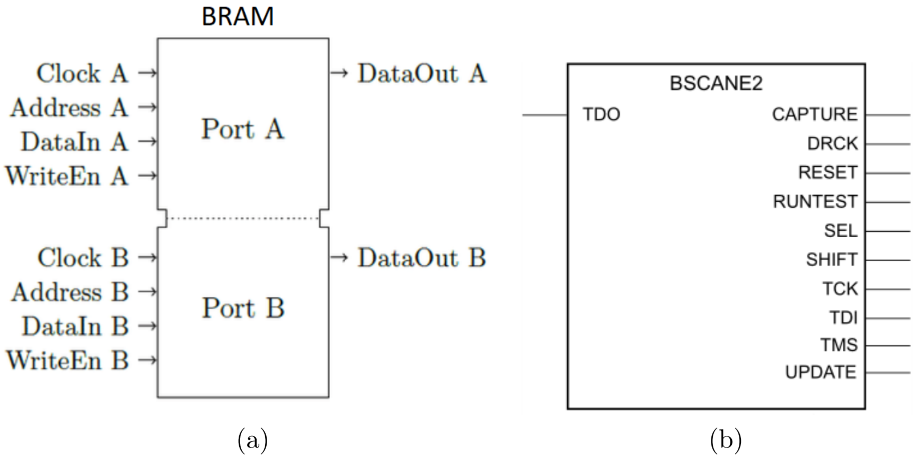
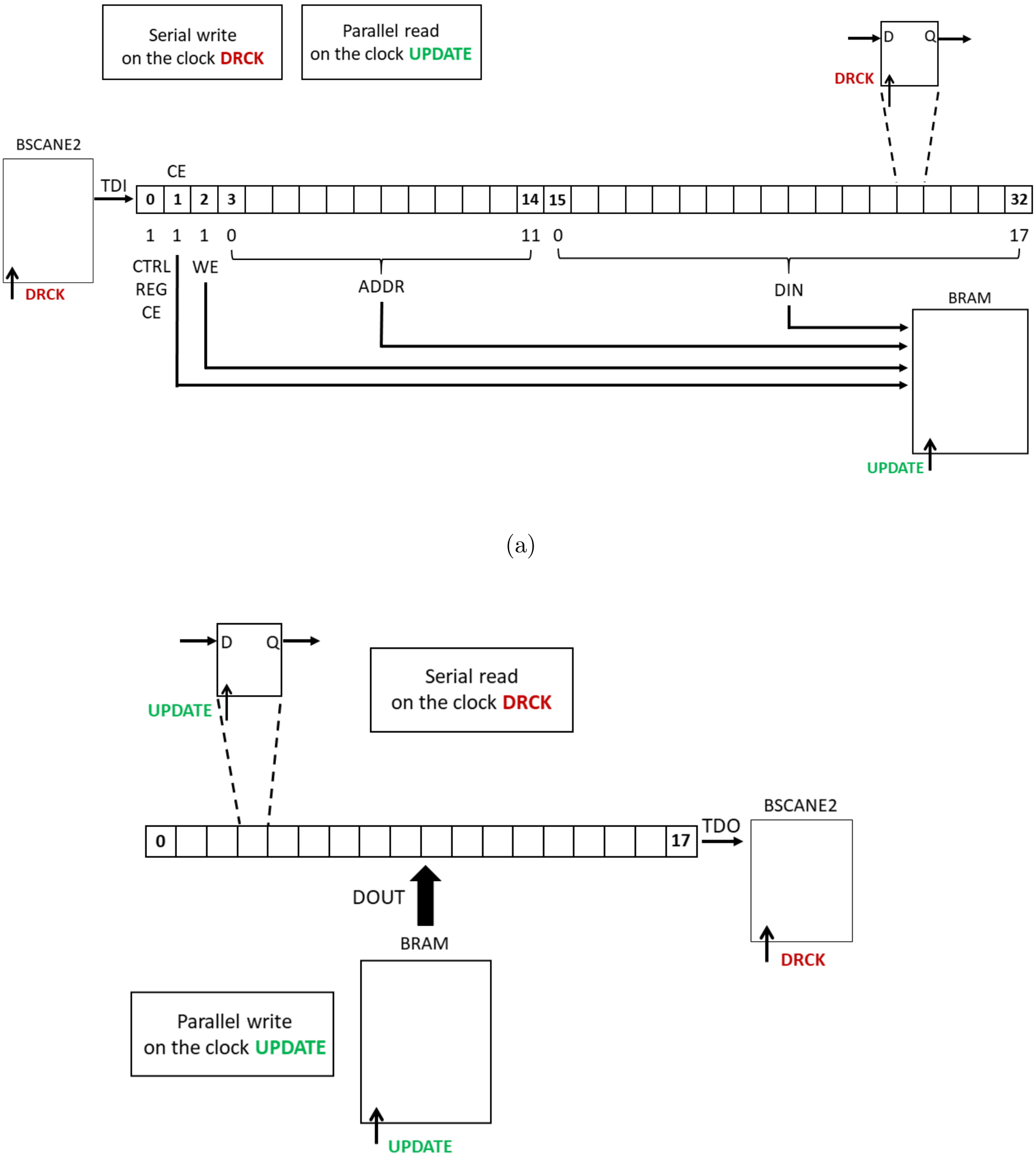


Figure 4.10: Scheme of the BRAM and BSCANE2 Xilinx primitive. (a) Representation of a dual port BRAM block scheme. (b) Representation of the BSCANE2 block scheme.

The JTAG Loader operates asynchronously with respect to the system clock, and two BSCANE2 signals are used to trigger the writing and reading operation of the BRAM content: DRCK and UPDATE. The former is the clock signal for the JTAG shifting data operation, whereas the latter indicates the end of the loading operation in the data register and represents the clock signal for the BRAM access operation [2]. During the JTAG Loader writing operation, a 33-stage shift-register is loaded at each DRCK rising edge. When the shifting operation is ended, the UPDATE signal switch to the high state. The BRAM reads in parallel the register content (BRAM ADDR, BRAM WE, BRAM EN, and BRAM DIN) on the UPDATE rising edge (see Fig. 4.11(a)). As far as the JTAG Loader reading operation is concerned, the BRAM writes in parallel a 18-stage shift-register on the UPDATE rising edge with the data outputs (BRAM DOUT). Then, the JTAG Loader serial reading of BRAM DOUT is performed on DRCK rising edge, as depicted in Fig 4.11(b).

[2] Details about the JTAG operation are available in [31].

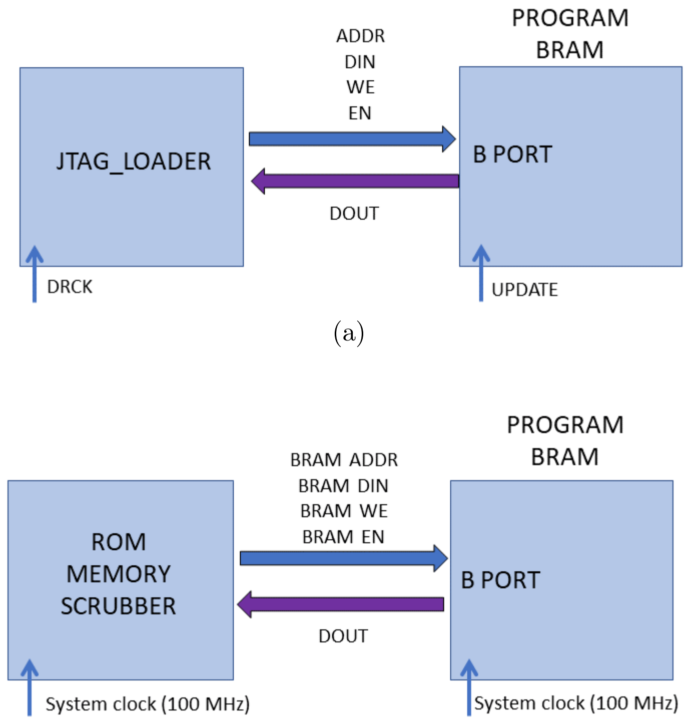


(a)

(b)

Figure 4.11: Schematic diagram of JTAG Loader writing and reading operations. (a) Representation of the writing operation to the BRAM. (b) Representation of the reading operation of the BRAM output data.

When the Xilinx JTAG Loader feature is used, the hardware majority voters and the corresponding memory scrubber cannot be used, since they are based on the same BRAM port (see Fig .4.12).



(a)

(b)

Figure 4.12: Representation of the previous Program BRAM possible communication with JTAG Loader and Program ROM memory scrubber. (a) JTAG Loader communication with the BRAM. (b) Program ROM scrubber communication with the BRAM.

In order to overcome these limitation, I modified the JTAG Loader structure in order to make it compatible to with the scrubbing of the Program ROM even when it is used. To this aim, I set two requirements:

1. a signal must identify which system is accessing the B port of the BRAM and multiplex inputs accordingly;

2. the JTAG Loader outputs must be synchronous with the system in order to avoid clock multiplexing on the B port.

As far as the first requirement is concerned, I have used the PB reset for such purpose, since the reset level is ’1’ for the PB during the uploading operation and switches to ’0’ at the end. Regarding the second requirement, I have envisaged a solution to synchronize the JTAG Loader outputs to the clock system. Regarding the second requirement, Indeed, the communication with the Program ROM is managed by a multiplexer(MUX), whose selection input is the PB reset, as shown in Fig. 4.13.

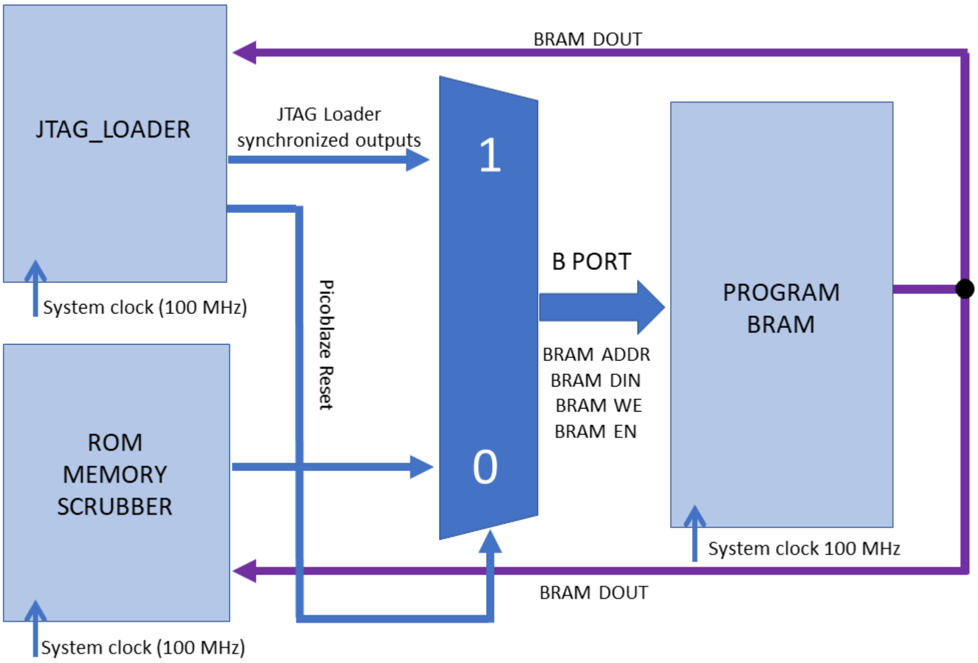


Figure 4.13: JTAG Loader architecture I proposed and implemented.

## 4.3 Software

As anticipated, the configuration readback and correction is performed in software by means of the PB microcontroller. In addition to the hardware development, I performed software modifications. Firstly, I have reorganized the the PB program - written in the PB assembler code (PSM) - in order to easily export the developed software on several devices. Indeed, I have split the PB Program in subcategories based on their functionality, and grouped in different files as described below (see Fig. 4.14).

- **Family.** Each FPGA family has a pertaining file which contains all the family-specific functionality. For instance, the detailed command sequences to access the configuration through ICAP have been splitted for 7-series family, Virtex-5 family, and also for Virtex-5 FPGAs scrubbing Spartan-6 FPGAs.

- **Device.** Each device has a file which includes all the device-specific functionality of a given family. For example, the IDCODE has been specified for the Kintex-7 70T and the Kintex-7 325T, whereas the IOB frame length, the number of device rows or column sizes have been included, for Spartan-6 LX45.

- **JTAG routines.** A file contains the assembly functions to access the configuration via the JTAG TAP Finite State Machine (FSM).

- **Program.** The main file includes all high-level functions for scrubbing the configuration including a terminal-like user interface.

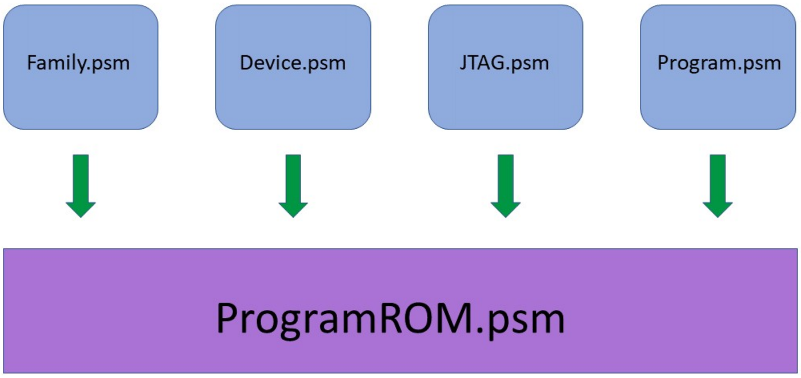


Figure 4.14: Schematic diagram of the software reorganization

The files related to family, device, and JTAG routines constitute libraries which must be included in the main Program file to obtain the Program ROM for the PB.

Moreover, I have added new functionalities to the previous C$^3$ circuit. Firstly, I have implemented a polling solution for the user-circuit communication in order to ensure that interruptions may occur only in specified phases of scrubbing operations.

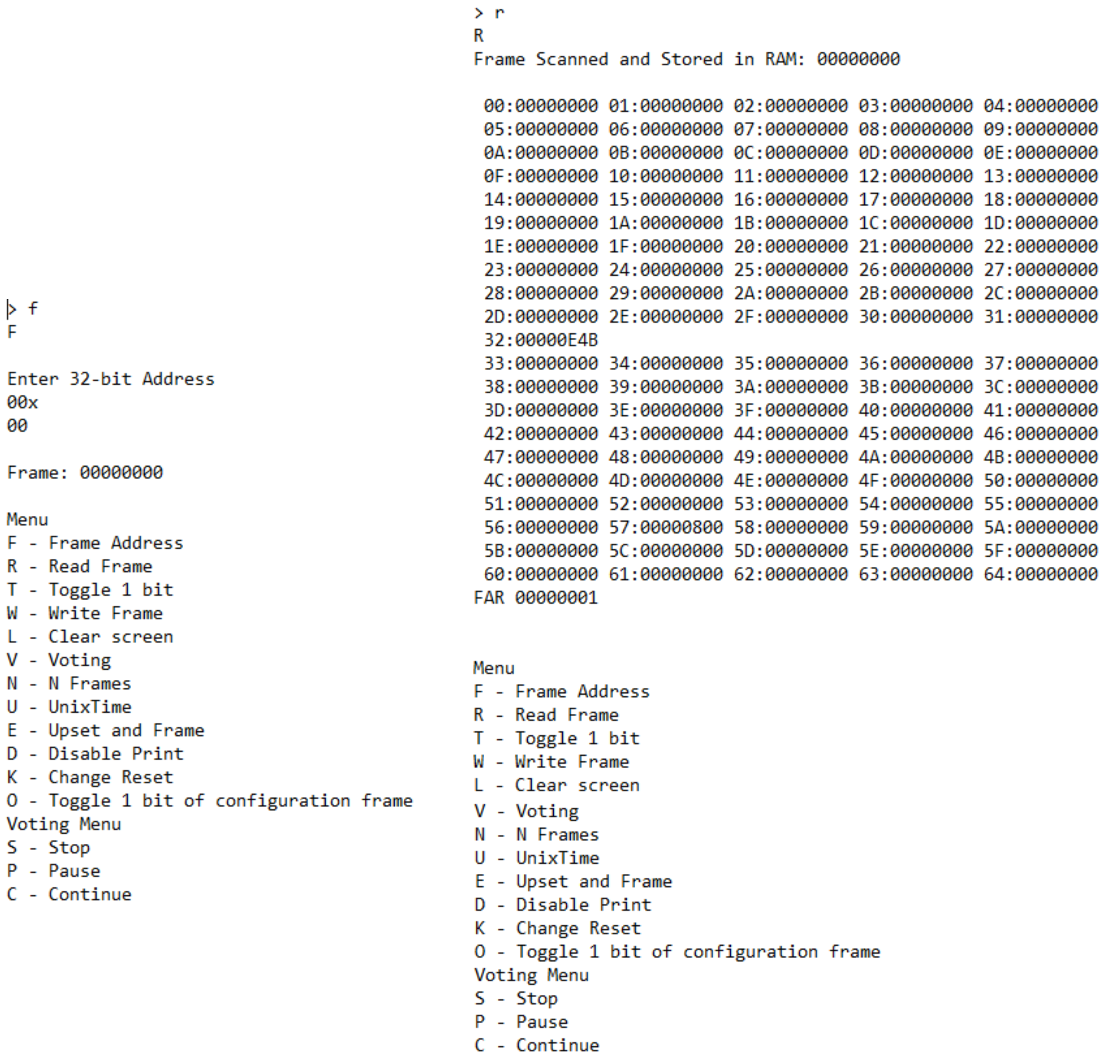
I also implemented additional functionality for the test phase. Indeed, to evaluate the C$^3$ reliability, "faults-injection" test have been performed. These tests consist in the toggling the configuration bits, in order to evaluate their effect on the circuit functionality. In the previous implementation, the toggle of a bit configuration was accomplished through four separated commands. These commands are conceived for the user, since each step is printed in order to show the produced changes in a configuration frame, and are described below.

- **F.** This command allows to set the frame address.
- **R.** This command returns the frame selected through F command.
- **T.** This command permits to toggle a bit in the frame indicated by F and returns the modified frame.
- **W.** This command writes the toggled frame in the configuration memory.

Each command produces several text lines of output as shown in Fig. 4.15 and Fig. 4.16, which are not ideal for the test phase, executed by means of automated TCL scripts.

In the actual implementation, in addition to the described commands, I have implemented a new single command (O), which sets the frame address, reads it, toggles the selected bit, and writes it to the configuration without output printing. The O command avoids cumbersome printouts for the test phase and reduces the data transit through the JTAG-UART interface, as shown in Fig. 4.17.

Furthermore, I have added a device protection check. It is possible that a single flipped bit indirectly generates several upsets on the other configuration bits. This effect can compromise the $C^3$ correction capability, which may affect the circuit functionality and also may damage the device. Therefore, I have implemented a specific check to halt the $C^3$ when the upsets in a single frame exceeded a chosen threshold. Moreover, it sends a message to inform the user of the circuit halt and that it needs reconfiguration.



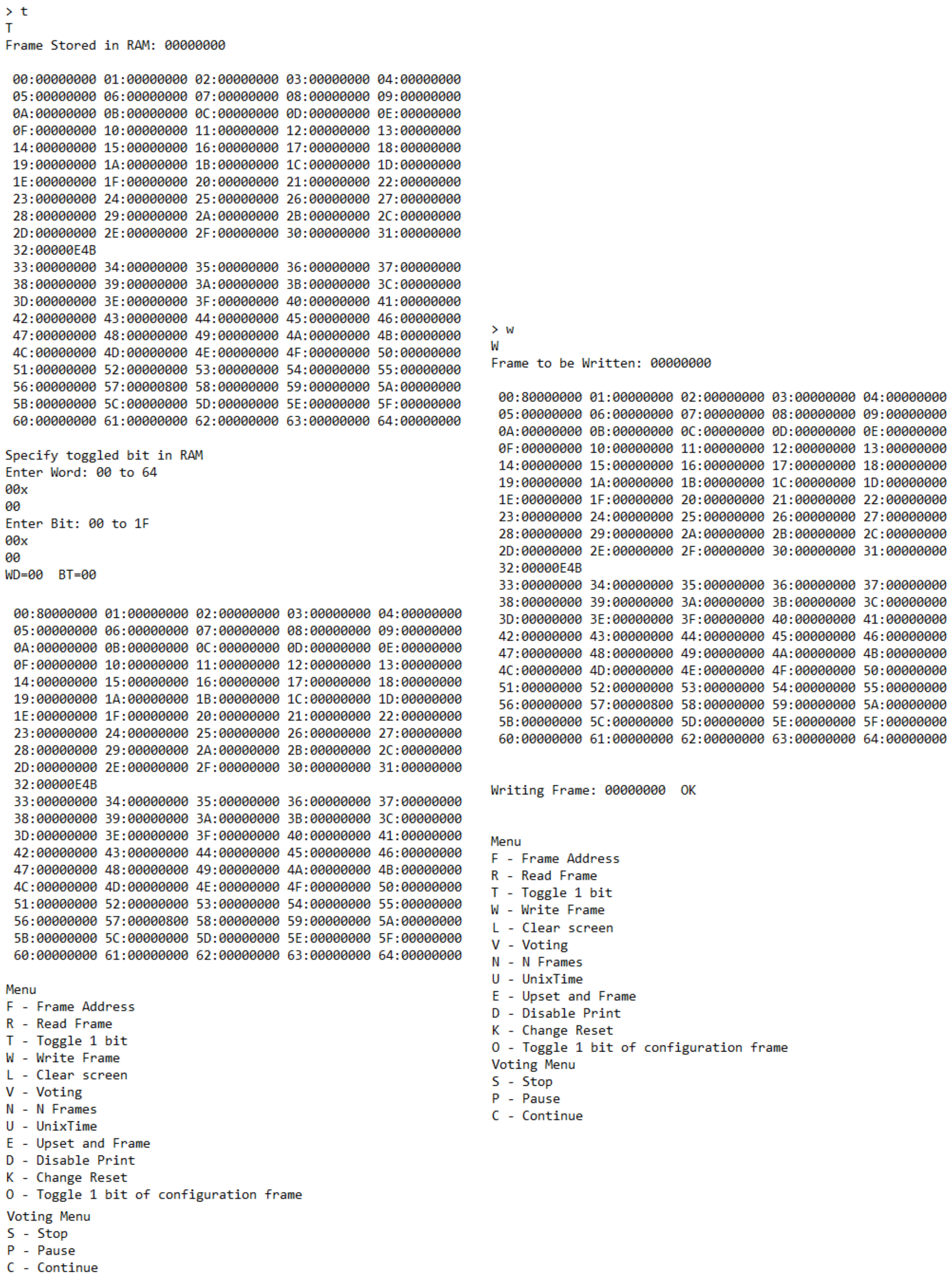
```
> f
F

Enter 32-bit Address
00x
00

Frame: 00000000

Menu
F - Frame Address
R - Read Frame
T - Toggle 1 bit
W - Write Frame
L - Clear screen
V - Voting
N - N Frames
U - UnixTime
E - Upset and Frame
D - Disable Print
K - Change Reset
O - Toggle 1 bit of configuration frame
Voting Menu
S - Stop
P - Pause
C - Continue
```

(a)

```
> r
R
Frame Scanned and Stored in RAM: 00000000

 00:00000000 01:00000000 02:00000000 03:00000000 04:00000000
 05:00000000 06:00000000 07:00000000 08:00000000 09:00000000
 0A:00000000 0B:00000000 0C:00000000 0D:00000000 0E:00000000
 0F:00000000 10:00000000 11:00000000 12:00000000 13:00000000
 14:00000000 15:00000000 16:00000000 17:00000000 18:00000000
 19:00000000 1A:00000000 1B:00000000 1C:00000000 1D:00000000
 1E:00000000 1F:00000000 20:00000000 21:00000000 22:00000000
 23:00000000 24:00000000 25:00000000 26:00000000 27:00000000
 28:00000000 29:00000000 2A:00000000 2B:00000000 2C:00000000
 2D:00000000 2E:00000000 2F:00000000 30:00000000 31:00000000
 32:00000E4B
 33:00000000 34:00000000 35:00000000 36:00000000 37:00000000
 38:00000000 39:00000000 3A:00000000 3B:00000000 3C:00000000
 3D:00000000 3E:00000000 3F:00000000 40:00000000 41:00000000
 42:00000000 43:00000000 44:00000000 45:00000000 46:00000000
 47:00000000 48:00000000 49:00000000 4A:00000000 4B:00000000
 4C:00000000 4D:00000000 4E:00000000 4F:00000000 50:00000000
 51:00000000 52:00000000 53:00000000 54:00000000 55:00000000
 56:00000000 57:00000800 58:00000000 59:00000000 5A:00000000
 5B:00000000 5C:00000000 5D:00000000 5E:00000000 5F:00000000
 60:00000000 61:00000000 62:00000000 63:00000000 64:00000000
FAR 00000001


Menu
F - Frame Address
R - Read Frame
T - Toggle 1 bit
W - Write Frame
L - Clear screen
V - Voting
N - N Frames
U - UnixTime
E - Upset and Frame
D - Disable Print
K - Change Reset
O - Toggle 1 bit of configuration frame
Voting Menu
S - Stop
P - Pause
C - Continue
```

(b)

Figure 4.15: Representation of the F and R command printouts used by the user to set the frame address and read the frame content, respectively. (a) F command printout. (b) R command printout.



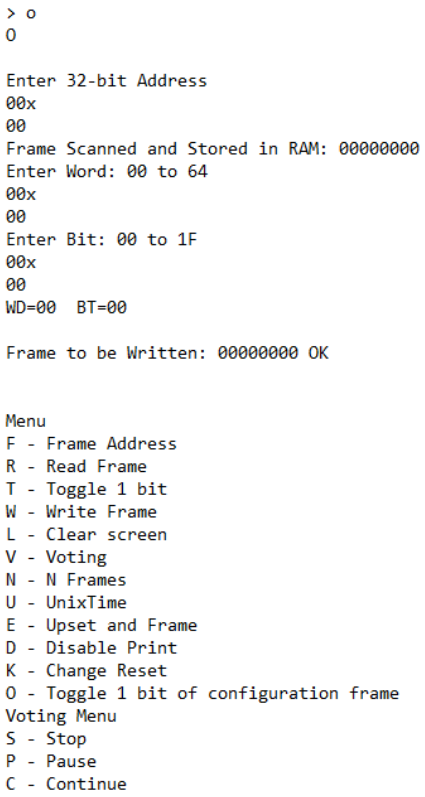
```
> t
T
Frame Stored in RAM: 00000000

 00:00000000 01:00000000 02:00000000 03:00000000 04:00000000
 05:00000000 06:00000000 07:00000000 08:00000000 09:00000000
 0A:00000000 0B:00000000 0C:00000000 0D:00000000 0E:00000000
 0F:00000000 10:00000000 11:00000000 12:00000000 13:00000000
 14:00000000 15:00000000 16:00000000 17:00000000 18:00000000
 19:00000000 1A:00000000 1B:00000000 1C:00000000 1D:00000000
 1E:00000000 1F:00000000 20:00000000 21:00000000 22:00000000
 23:00000000 24:00000000 25:00000000 26:00000000 27:00000000
 28:00000000 29:00000000 2A:00000000 2B:00000000 2C:00000000
 2D:00000000 2E:00000000 2F:00000000 30:00000000 31:00000000
 32:00000E4B
 33:00000000 34:00000000 35:00000000 36:00000000 37:00000000
 38:00000000 39:00000000 3A:00000000 3B:00000000 3C:00000000
 3D:00000000 3E:00000000 3F:00000000 40:00000000 41:00000000
 42:00000000 43:00000000 44:00000000 45:00000000 46:00000000
 47:00000000 48:00000000 49:00000000 4A:00000000 4B:00000000
 4C:00000000 4D:00000000 4E:00000000 4F:00000000 50:00000000
 51:00000000 52:00000000 53:00000000 54:00000000 55:00000000
 56:00000000 57:00000800 58:00000000 59:00000000 5A:00000000
 5B:00000000 5C:00000000 5D:00000000 5E:00000000 5F:00000000
 60:00000000 61:00000000 62:00000000 63:00000000 64:00000000

Specify toggled bit in RAM
Enter Word: 00 to 64
00x
00
Enter Bit: 00 to 1F
00x
00
WD=00  BT=00

 00:80000000 01:00000000 02:00000000 03:00000000 04:00000000
 05:00000000 06:00000000 07:00000000 08:00000000 09:00000000
 0A:00000000 0B:00000000 0C:00000000 0D:00000000 0E:00000000
 0F:00000000 10:00000000 11:00000000 12:00000000 13:00000000
 14:00000000 15:00000000 16:00000000 17:00000000 18:00000000
 19:00000000 1A:00000000 1B:00000000 1C:00000000 1D:00000000
 1E:00000000 1F:00000000 20:00000000 21:00000000 22:00000000
 23:00000000 24:00000000 25:00000000 26:00000000 27:00000000
 28:00000000 29:00000000 2A:00000000 2B:00000000 2C:00000000
 2D:00000000 2E:00000000 2F:00000000 30:00000000 31:00000000
 32:00000E4B
 33:00000000 34:00000000 35:00000000 36:00000000 37:00000000
 38:00000000 39:00000000 3A:00000000 3B:00000000 3C:00000000
 3D:00000000 3E:00000000 3F:00000000 40:00000000 41:00000000
 42:00000000 43:00000000 44:00000000 45:00000000 46:00000000
 47:00000000 48:00000000 49:00000000 4A:00000000 4B:00000000
 4C:00000000 4D:00000000 4E:00000000 4F:00000000 50:00000000
 51:00000000 52:00000000 53:00000000 54:00000000 55:00000000
 56:00000000 57:00000800 58:00000000 59:00000000 5A:00000000
 5B:00000000 5C:00000000 5D:00000000 5E:00000000 5F:00000000
 60:00000000 61:00000000 62:00000000 63:00000000 64:00000000

Menu
F - Frame Address
R - Read Frame
T - Toggle 1 bit
W - Write Frame
L - Clear screen
V - Voting
N - N Frames
U - UnixTime
E - Upset and Frame
D - Disable Print
K - Change Reset
O - Toggle 1 bit of configuration frame
Voting Menu
S - Stop
P - Pause
C - Continue
```

(a)

```
> w
W
Frame to be Written: 00000000

 00:80000000 01:00000000 02:00000000 03:00000000 04:00000000
 05:00000000 06:00000000 07:00000000 08:00000000 09:00000000
 0A:00000000 0B:00000000 0C:00000000 0D:00000000 0E:00000000
 0F:00000000 10:00000000 11:00000000 12:00000000 13:00000000
 14:00000000 15:00000000 16:00000000 17:00000000 18:00000000
 19:00000000 1A:00000000 1B:00000000 1C:00000000 1D:00000000
 1E:00000000 1F:00000000 20:00000000 21:00000000 22:00000000
 23:00000000 24:00000000 25:00000000 26:00000000 27:00000000
 28:00000000 29:00000000 2A:00000000 2B:00000000 2C:00000000
 2D:00000000 2E:00000000 2F:00000000 30:00000000 31:00000000
 32:00000E4B
 33:00000000 34:00000000 35:00000000 36:00000000 37:00000000
 38:00000000 39:00000000 3A:00000000 3B:00000000 3C:00000000
 3D:00000000 3E:00000000 3F:00000000 40:00000000 41:00000000
 42:00000000 43:00000000 44:00000000 45:00000000 46:00000000
 47:00000000 48:00000000 49:00000000 4A:00000000 4B:00000000
 4C:00000000 4D:00000000 4E:00000000 4F:00000000 50:00000000
 51:00000000 52:00000000 53:00000000 54:00000000 55:00000000
 56:00000000 57:00000800 58:00000000 59:00000000 5A:00000000
 5B:00000000 5C:00000000 5D:00000000 5E:00000000 5F:00000000
 60:00000000 61:00000000 62:00000000 63:00000000 64:00000000


Writing Frame: 00000000  OK


Menu
F - Frame Address
R - Read Frame
T - Toggle 1 bit
W - Write Frame
L - Clear screen
V - Voting
N - N Frames
U - UnixTime
E - Upset and Frame
D - Disable Print
K - Change Reset
O - Toggle 1 bit of configuration frame
Voting Menu
S - Stop
P - Pause
C - Continue
```

(b)

Figure 4.16: Representation of the T and W command printouts used by the user to toggle a bit in a configuration frame and write it in the configuration, respectively. Note that the frame words format is big-endian (a) T command printout showing the previous and the new configuration frame. (b) W command printout.



```
> o
O

Enter 32-bit Address
00x
00
Frame Scanned and Stored in RAM: 00000000
Enter Word: 00 to 64
00x
00
Enter Bit: 00 to 1F
00x
00
WD=00  BT=00

Frame to be Written: 00000000 OK


Menu
F - Frame Address
R - Read Frame
T - Toggle 1 bit
W - Write Frame
L - Clear screen
V - Voting
N - N Frames
U - UnixTime
E - Upset and Frame
D - Disable Print
K - Change Reset
O - Toggle 1 bit of configuration frame
Voting Menu
S - Stop
P - Pause
C - Continue
```

Figure 4.17: Picture of the O command printout.

## 4.4 Plain C$^3$ implementation

The C$^3$ circuit has been designed for the INFN test board which hosts a XC7K325T FPGA used in BEAST II. Since no equal boards were available in Naples, where I have developed the circuit, I used a commercial off-the-shelf Xilinx KC705 board, that is shown in Fig. 4.18(a), and hosts the same BEAST II FPGA. The KC705 board provides a fixed oscillator with a differential 200 MHz output (which I have used to determine the desired DCO frequency), an UART to USB bridge, a JTAG configuration circuitry to enable configuration over USB. The board also includes several power converters transforming a 12V DC power supply inputs to the required voltages for the components on-board, including the FPGA. and a 12 V wall adapter to provide the required power supply to the FPGA.

The FPGA has been programmed with the C$^3$ circuit described in Sec. 4.2, and

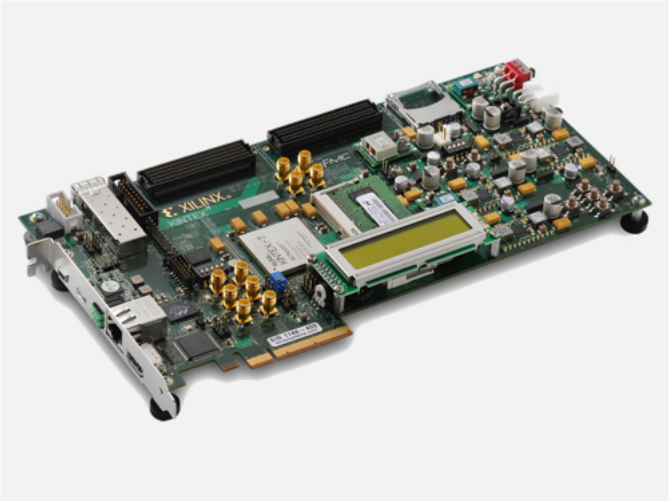
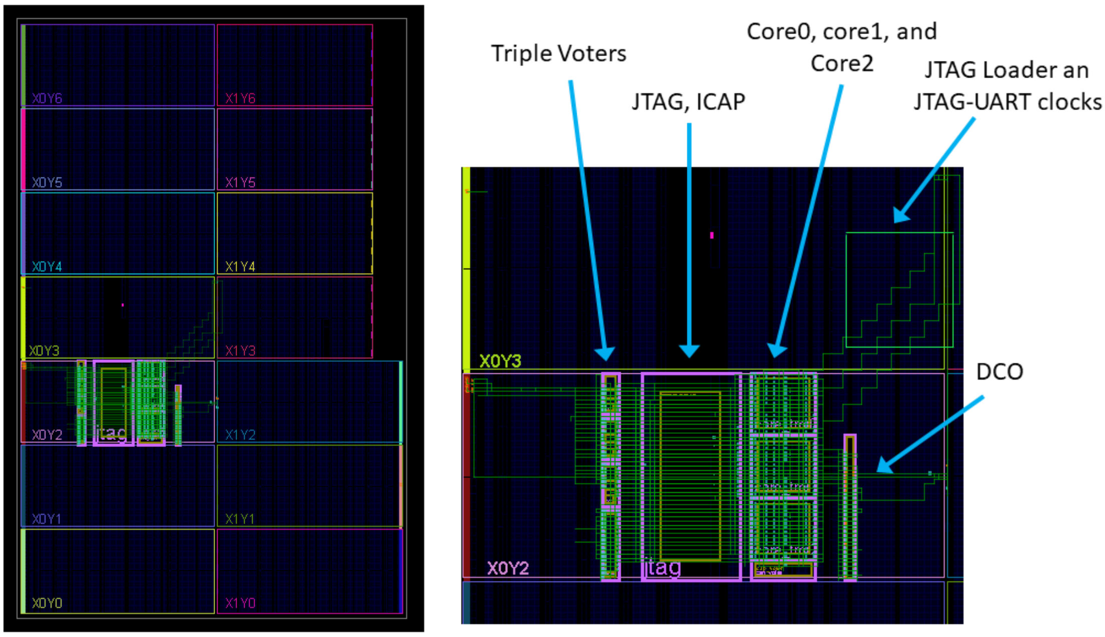


Figure 4.18: Schematic of KC705 board.

the circuit has been implemented by means of a manual floor-planning. In general, floorplanning is the process of geometrically constraining logic in a blocks (pblocks), and of manually placing them in the fabric, in order to increase density, routability or performances. The intent is normally to reduce route delays or area for the selected logic by optimizing placement. Since the proposed scrubber method is based on the redundancy of the configuration, I used the floorplanning to restrict the layout to a specific area. Indeed, the configuration confinement simplifies the configuration processing needed for C$^3$ operation. In this first implementation the whole circuit occupies approximately a single clock region, except for three clock signals required for JTAG Loader and JTAG-UART features, as shown in Fig. 4.19(b).

Regarding the impact of the logic utilization on the configuration utilization, placement and routing have been constrained to be in regions as vertically-narrow as possible, since configuration frames run vertically in the device. In this way, the number of configured frames for a given logic is minimized. In terms of utilized resources, the Table 4.1 shows the number of used and available resources, and their utilization percentage. In particular, ICAPE2 and BSCANE2 are the primitives to access the configuration through ICAP and for JTAG features, respectively. Moreover, in 4.1, are presented two buffer types: BUFG and BUFH. The former is used for global clocks, i.e. it can reach logic throughout the entire device, whereas the latter is a local clock buffer since it drives a horizontal global clock tree spine in a



(a) (b)

Figure 4.19: Layout of the plain C$^3$ implementation in a Xilinx 7K325T device. (a) Representation of implemented circuit: the routing is depicted as green paths, whereas pink squares represent pblocks. (b) Close-up of the implemented circuit.

single region (row). Regarding the maximum clock frequency for this circuit, it can be calculated basing on the Worst Negative Slack (WNS). The slack is defined as the time difference between the maximum time of propagation of a signal through a specific path based on the design requirements, and its estimated arrival. To satisfy the circuit setup constraints, the slack must be a positive value, i.e. a signal should reach its destination through the designated path before its maximum time of arrival. Therefore, a negative value of the WNS expresses the presence of one or more timing errors, i.e. it indicates that the set frequency is too high for the designed circuit. Instead, a positive WNS indicates that the timing requirements are satisfied and it is possible to utilize a higher clock frequency. In this implementation, the obtained WNS is 2.179 ns and the maximum reachable clock frequency which satisfies the timing budget is $\nu_{max} = \frac{1}{(10-2.179)}$ ns$^{-1} \simeq$ 128 MHz.

As previously mentioned, in this thesis work I have realized two C$^3$ implementations. The second implementation has been developed by means of the Xilinx

Table 4.1: Logic circuit resources occupation for the plain C$^3$ implementation.

| **Logic Resources** | **Available** | **Used** | **%** |
|---|---|---|---|
| Slices: LUT | 203800 | 2343 | 1.15 |
| Slices: FFs | 407600 | 2333 | 0.57 |
| Slices: overall | 50950 | 911 | 1.79 |
| F7 muxes | 101900 | 60 | 0.06 |
| F8 muxes | 50950 | 24 | 0.05 |
| BUFGCs | 32 | 3 | 9.37 |
| BUFHs | 168 | 2 | 1.19 |
| BSCANE2 | 4 | 2 | 50.00 |
| RAMB36 | 445 | 9 | 2.02 |

Isolation Design Flow (IDF) with Vivado and performing a triplication of the C$^3$ global signals. Before illustrating the second implementation, I will introduce the Xilinx IDF.

## 4.5 Isolation design flow

The IDF technique is utilized in design applications as information assurance (single chip cryptography), avionics, automotive, and other industrial applications [38]. Its basic idea is the isolation of the logic functions, in order to avoid that the alteration of a logic block functionality may alter the functionality of the other ones. The use of the IDF in a circuit design requires to define some concepts, described below.

- **Logical ownership.** Each logic component, described through an HDL module, logically owns the logic instantiated within it.
- **Physical ownership.** Each logic component can be used if it is physically owned by an isolated region.
- **Fence.** A fence is a set of unused tiles, i.e. lacking of routing and logic.
- **Function.** A function is collection of logic that performs a specific operation.
- **Trusted Routing.** The trusted routing is a subset of available routing resources meeting some restrictions depicted in Fig. 4.20. These restrictions

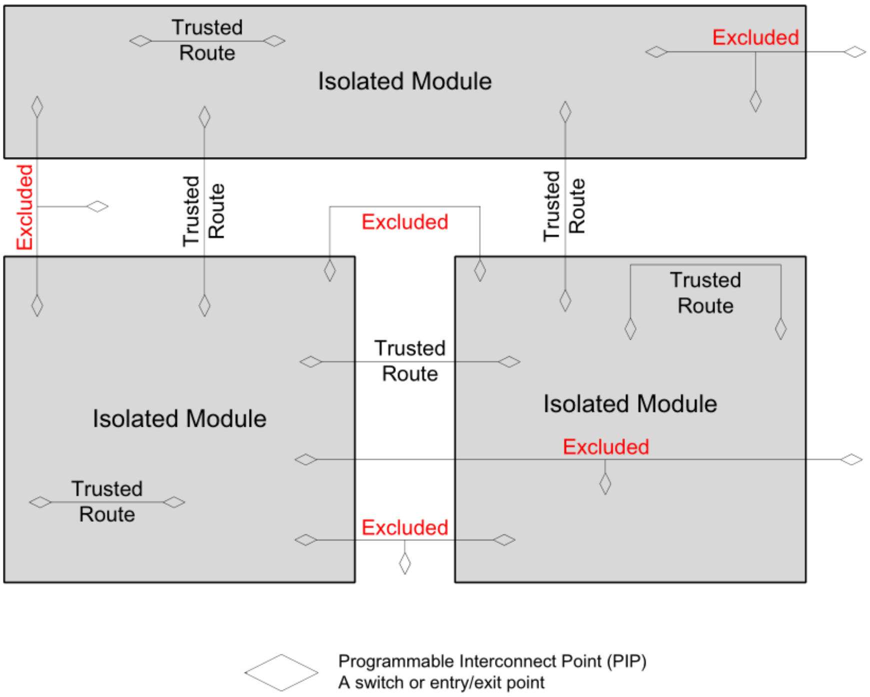


Figure 4.20: Representation of authorized and denied trusted routing paths between isolated regions.

include the absence of PIP in a fence between isolated regions( i.e. IDF avoid the routing "touchdowns"), the complete containment of the routing in source or destination Pblocks, and the constraint for which one source has only one destination.

IDF development requires to consider floorplanning at beginning phase of design process, since the development flow is based on hierarchy, i.e. each function to be isolated must corresponds to an isolated region. Therefore, a correspondence between logical and physical ownership is needed. Moreover, when the isolated modules have been defined, a fence must be used to separate them, i.e. for each defined component corresponding to an isolated module. In this way, all the logic included in lower hierarchy levels needed for the component description, will be isolated in the specified Pblock.

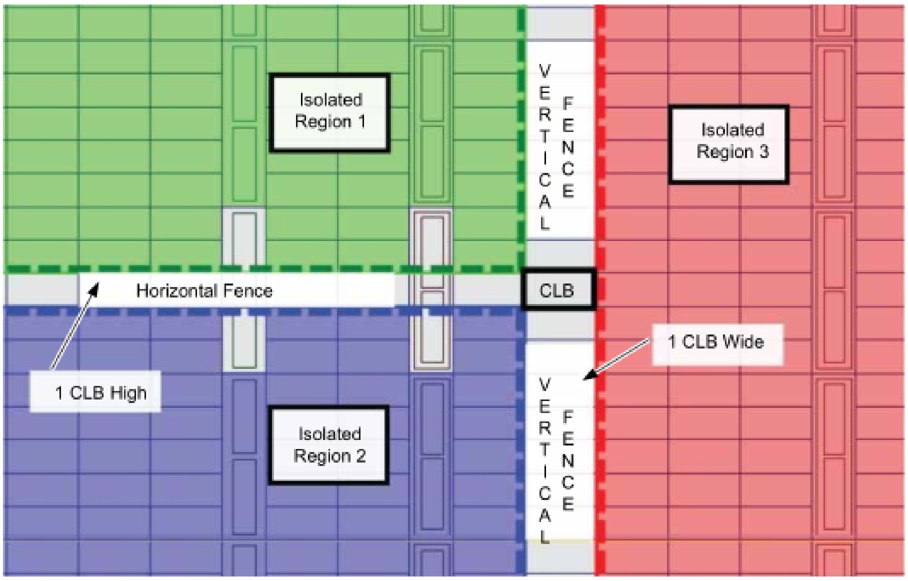


Figure 4.21: Representation of CLBs resources fence with an example of CLBs fences in horizontal and vertical directions.

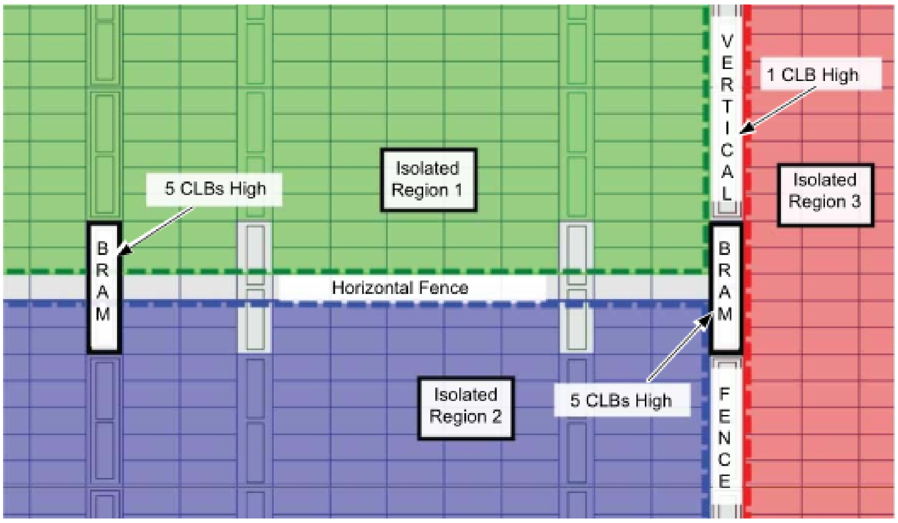


Figure 4.22: Representation of BRAMs resources fence with an example of BRAMs fences in horizontal and vertical directions.

In a typical IDF design the global logic must be instantiated at the top level of hierarchy and Xilinx recommend that it is only the clock logic. However, any component that is not part of an isolated module in the design hierarchy is moved to the top level. Furthermore, the global logic will be placed automatically by the Vivado tool, which ensures that it will not be placed in the fence. Anyway, it is possible to place the top level logic in an isolated region as long as the global logic will be exempt from isolation, by using the appropriate constraints. As far as the routing is concerned, there are no restrictions on top level logic other than it will not violate the fence with used PIPs. The fence definition depends on the resources to which it refers as shown in Fig. 4.21, and in Fig. 4.22 and must be defined both in horizontal and vertical direction.

## 4.6 IDF C$^3$ implementation

In this implementation, I have modified the C$^3$ architecture (Sec. 4.2) in order to fulfil the IDF requirements, as illustrated in Fig. 4.23. Firstly, I have modified Data RAM, Program ROM, scratchpad, reset voters, and their communication with the three cores. Indeed, in the previous architecture, for each of them there was a single component consisting of three voters and communicating with the three cores. Here, the three cores communicate with three Logic TMR, each one consisting of a single Data RAM, Program ROM, scratchpad and reset voter. Therefore, in this case voters are grouped referring to the cores rather than the kind of voted signals. This modification arises from two observations:

1. in order to preserve the C$^3$ reliability, a damage on a single TMR module must not affect the remaining ones (avoid any single point of failures);
2. due to the IDF rules based on the logical and physical ownership correspondence, it is not possible to instantiate an isolated module merging different hierarchy levels.

Note that the first principle was already satisfied in the first implementation, where no logical and physical ownership correspondence was needed. Since the IDF prevents routing touchdowns in the fence and each source must have a single destination, I have designed a new component, indicated as MIXER in Fig. 4.23, to make communication possible between the three Logic TMR blocks with the three cores, taking into account the fan-out of each signal. Furthermore, in this implementation, each Data RAM and Program ROM memory scrubber become parts of the corresponding core.

As far as the BSCANE2 and ICAPE2 primitives are concerned, since their logic is global, they belong to the same non-isolated pblock, which is named as configuration block. Therefore, in order to avoid that the logic paths between configuration block and cores represent a single point of failure for the circuit, I have also tripled the fan-out for all the BSCANE2 and ICAPE2 primitives signals which were destined to the cores. I have tripled also the clock signal by using three different clock buffers, i.e. each core has its clock.

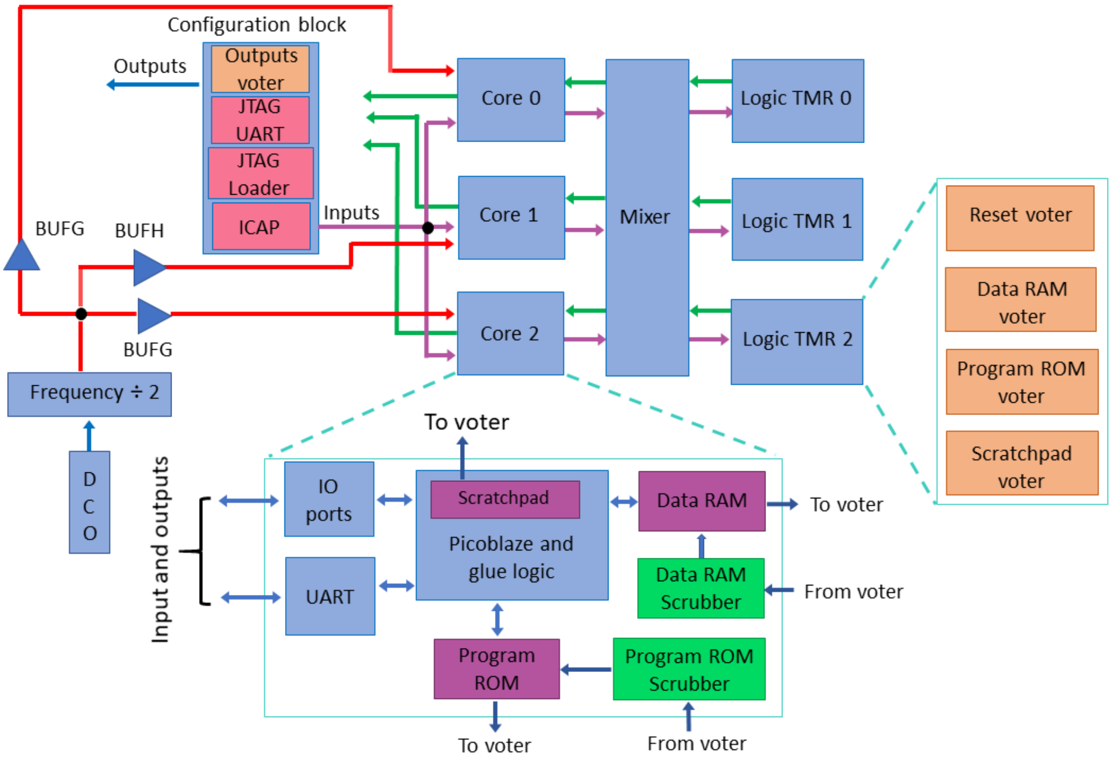
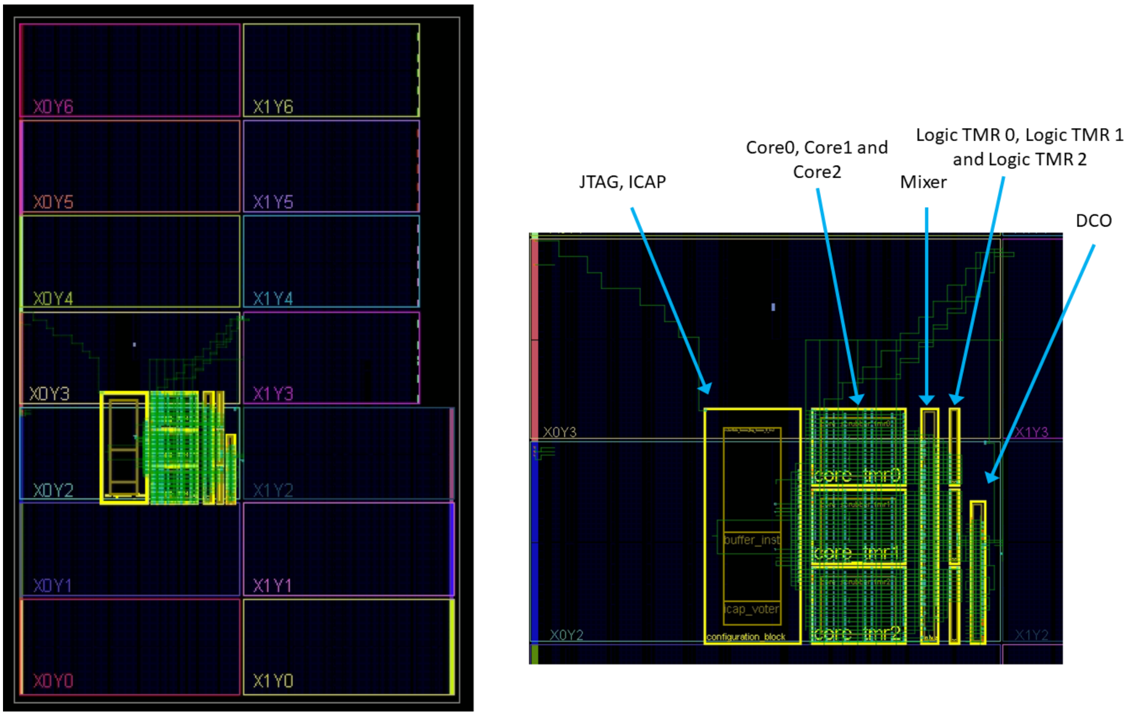


Figure 4.23: Schematic diagram the architecture $C^3$ IDF implementation.

Due to the IDF requirements, the new implementation occupies a larger area in the fabric, as shown in Fig. 4.24, and the DCO placement is bonded closer the clock buffer, as anticipated in Sec. 4.2.1.



(a) (b)

Figure 4.24: Layout of the IDF $C^3$ implementation. (a) Representation of implemented circuit: the routing is depicted as green paths, whereas yellow squares represent pblocks. (b) Close-up of the implemented circuit.

Table 4.2: Logic circuit resources occupation for the IDF and the plain $C^3$ implementation.

| **Logic Resources IDF** | **Available** | **Used IDF** | **% IDF** | **Used Plain** | **% Plain** |
|---|---|---|---|---|---|
| Slices: LUT | 203800 | 3534 | 1.73 | 2343 | 1.15 |
| Slices: FFs | 407600 | 2771 | 0.68 | 2333 | 0.57 |
| Slices: overall | 50950 | 1328 | 2.60 | 911 | 1.79 |
| F7 muxes | 101900 | 54 | 0,06 | 60 | 0.06 |
| F8 muxes | 50950 | 24 | 0.05 | 24 | 0.05 |
| BUFGs | 32 | 8 | 25 | 3 | 9.37 |
| BUFHs | 168 | 1 | 0.59 | 2 | 1.19 |
| BSCANE2 | 4 | 2 | 50.00 | 2 | 50 |
| RAMB36 | 445 | 9 | 2.02 | 9 | 2.02 |

In this implementation, the WNS is 2.630 ns, so that the maximum reachable frequency for the circuit is $\nu_{max} = \frac{1}{(10-2.630)}$ ns$^{-1}\simeq$ 136 MHz, that is an higher value with respect to the plain implementation where the maximum reachable frequency was $\simeq$ 128 MHz. Table 4.2 shows the logic resources used for the IDF and the plain C$^3$ implementations.

Note that the occupancy of two clock regions and the triplication of the clock signal have increased the number of BUFGs, whereas the number of BUFH has been halved. Moreover the LUT used in IDF implementation is significantly increased whereas the number of FFs is decreased. In the next chapter, the fault injection test results for the plain and IDF C$^3$ implementation will be discussed.

# Chapter 5

# Fault injection Test Results

In this chapter the tests for the evaluation of the C$^3$ self-repair capability is discussed. Moreover, test circuits devised to explore the IDF and its benefits in terms of reliability are presented.

## 5.1 Evaluation of the Xilinx isolation design flow on test circuits

The second C$^3$ implementation has been realized on the basis of the results of the IDF impact on the test circuits. These circuits have the same basic structure and differ only for the presence or absence of the IDF constraints and the floorplan implementation. I have evaluated their reliability by comparing results of fault injection tests, consisting in a sequential injection of errors, performed by toggling the circuit essential bits (defined in Sec. 3.3.4).

The circuit structure is represented in Fig 5.1 and consists of four 32-bit counters (C0, C1, C2, and CREF), some glue logic, and a VIO. The counters are clocked on the 200 MHz clock on the KC705 board described in Sec. 4.4. I have designed the glue logic to transfer C0, C1, and C2 outputs to the VIO, which allows someone to read the counters value during the test and is excluded from the fault injection. Also CREF is excluded from the injection, since it is uses as a count reference. Therefore, CREF and VIO are placed in a different area of the fabric with respect to C0, C1, C2, and the glue logic, which are instead test targets.

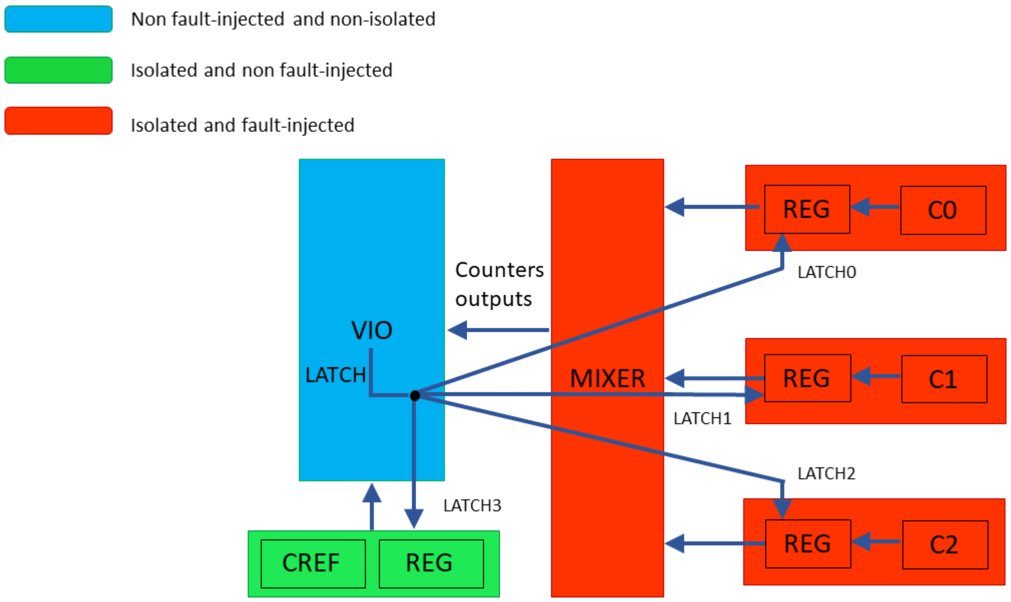

Non fault-injected and non-isolated
Isolated and non fault-injected
Isolated and fault-injected
VIO
LATCH
Counters outputs
MIXER
REG
C0
LATCH0
REG
C1
LATCH1
LATCH2
REG
C2
LATCH3
CREF
REG

Figure 5.1: Architecture of test circuits.

I have designed the glue logic, Mixer in Fig. 5.1, both to test its performance for the IDF implementation of the C$^3$ and to merge paths between the three tested counters, in order to evaluate possible differences between the plain and the IDF implementation. Moreover, since the VIO logic is global, I have exempted it from the isolation, i.e. the VIO belongs to a non-isolated pblock. The identification of the circuit essential bits (defined in 3.3.4) for each implementation is performed through a TCL script after the bitstream generation. Moreover, by means of an academic software designed at the Department of Physics, named ConfigView, it is possible to identify the C0, C1, C2 frame addresses, in order to exclude the others from the injection list. Indeed, ConfigView opens the readback file, obtained through a specific TCL script, and produces a configuration bit map, indicating the programmed bits and the empty ones with different colours. Therefore, this software allows to visually compare the FPGA readback configuration to the configured modules observed through the circuit layout. Moreover, it can also overlap a chosen bit-list to the readback plot with another colour, in order to highlight a specific category of bits. Once the essential bits for the injection have been selected, the FPGA is programmed with the bitstream and the fault injection test can be executed by means of the fault-injection script via JTAG port. Such a script also performs voting and comparison operations, which are, therefore, preserved from the test corruption. As far as the test structure is concerned, I have organized it in three steps.

1. **Sample generation.** A subset of the essential bits is randomly extracted as a sample.
2. **Single bit injection.** The bits of the sample are sequentially toggled (one by one) in the configuration.
3. **Comparison.** At each error injection, C0, C1, C2 values are voted and the resulting vote is compared to the CREF value. When a mismatch between voted and expected value is found, the toggled bit is reported as critical, i.e. it causes a failure of the circuit.
4. **Correction.** After each injection, the flipped bit is again corrected. If a counter does not recover its correct operation after this partial reconfiguration, the FPGA is fully reprogrammed. After a partial or full reconfiguration, the test restarts from the second step.

The JTAG serial frequency bit scan is 12 Mb whereas the VIO reads 256 Mb in parallel, i.e. the VIO reading frequency is $\simeq$ 21 times higher than the JTAG one. As a consequence, the script also manages a VIO input port (**LATCH**) signal, to latch the counts into registers before the comparison operation (see Fig. 5.1).

In the test circuits that I have conceived, a critical bit is such if itthe simultaneous failure of at least two counters, in order to evaluate the IDF impact on the circuit. Indeed, by definition, the isolation must guarantee that a failure in a module does not influence the functionality of the others.

For each circuit version, with and without IDF, I have designed an additional version with tripled global signals. In the next section, I will present and analyse the tests results on these circuits.

## 5.2 IDF evaluation test results

For testing the designed circuits, the Kintex-7 325T on the KC705 board described in 4.4 has been used, and access the configuration has been performed via JTAG port.

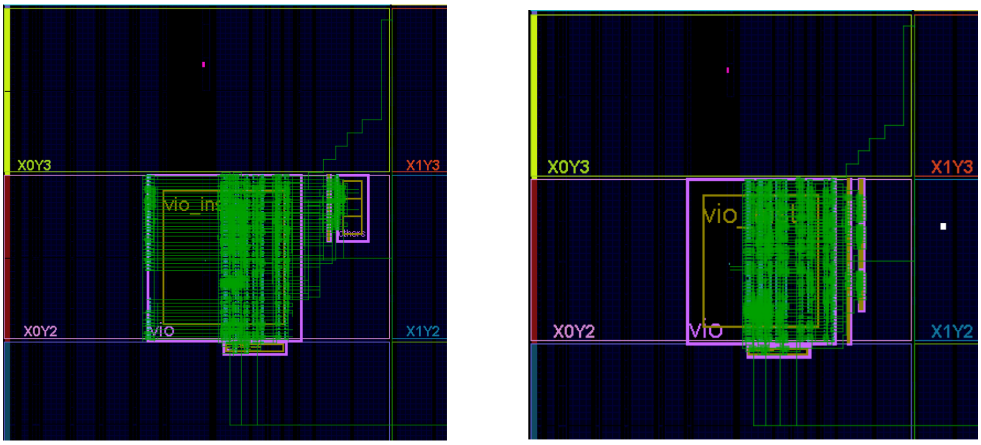
As mentioned in Sec. 4.5, the global logic is exempted from isolation in an IDF circuit implementation, i.e. these signals do not need to comply to the requirements of lower logic hierarchical levels. For instance, the clock signal must communicate with all the logical blocks in order to ensure the system synchronization. Therefore, on the basis of the IDF rule for trusted routing, I have tripled the global signals to maintain the single source-destination rule also for the global signals. To this aim, I have instantiated a clock buffer for each counter and tripled the **LATCH** signal to the counters output latch. As a consequence, four test circuits have been implemented. The implementations of the plain and the IDF counters are depicted in Fig. 5.2, whereas the implementations with global logic tripled are not reported since their layout are comparable to the previous ones.

**a** the plain implementation;

**b** IDF implementation;

**c** plain implementation with triple global signals;

**d** IDF implementation with triple global signals.



(a) (b)

Figure 5.2: Layout of the two counters circuits implementation without the global signals tripled. (a) Plain circuit implementation. (b) IDF counters versions.

Regarding the test results analysis, I have evaluated the impact of IDF and the triplication of the global signals by comparing the fraction of critical bits over the corrupted essential bits for each circuit test. At each fault injection two possible situations may occur with a constant probability, since the corrupted bit may cause the circuit failure or not. As a consequence, the analysis of the tests results can be performed by using the binomial distribution. Given $n$ independent Bernoulli trials (trials taking the value 1 for a success with probability $p$ and 0 for a failure with probability $q = 1 - p$), the binomial distribution $B(k; n, p)$ gives the probability of getting exactly $k$ successes. Therefore, if $X$ is the random variable following the binomial distribution, $X \sim B(X; n, p)$, such a probability is $B(X = k; n, p) = \binom{n}{k} p^k q^{n-k}$. The mean value and the variance of the binomial distribution are

$$\mu_X = np, \tag{5.1a}$$

$$\sigma_X^2 = npq, \tag{5.1b}$$

respectively. In these tests, it is possible to measure the number of success as the number of found critical bits. Therefore, starting from a randomly obtained sample of the circuit essential bits, it is possible to refer to the distribution of the sample proportion $\hat{p}$ to estimate the probability $p$ of finding a critical bit within the essential bits population of the considered circuit [39]. Since for a sample size $n$ the sample proportion is defined as $\hat{p} = X/n$, the $\mu_{\hat{p}}$ and $\sigma_{\hat{p}}^2$ are equal to

$$\mu_{\hat{p}} = \mu(X/n) = \frac{\mu(X)}{n} = \frac{np}{n} = p, \tag{5.2a}$$

$$\sigma_{\hat{p}}^2 = \sigma^2(X/n) = \frac{np(1-p)}{n^2} = \frac{p(1-p)}{n}. \tag{5.2b}$$

The first equation proves that the sample proportion is an unbiased estimator of the population proportion, i.e. of the probability $p$, whereas the second indicates that variance decreases as the size of the sample increases. Fig. 5.3 and 5.4 show the geometric distributions of the essential bits in the device for the plain implementation, respectively without and with the triplication of the global signals. Such distributions are represented in Fig. 5.5 and 5.6 for the IDF implementations. Note that for each test, I have used an essential bits sample size of 7200 which represents a compromise between the test duration (ten of hours) and the error on the estimator $\epsilon_{\hat{p}} \simeq 20\%$

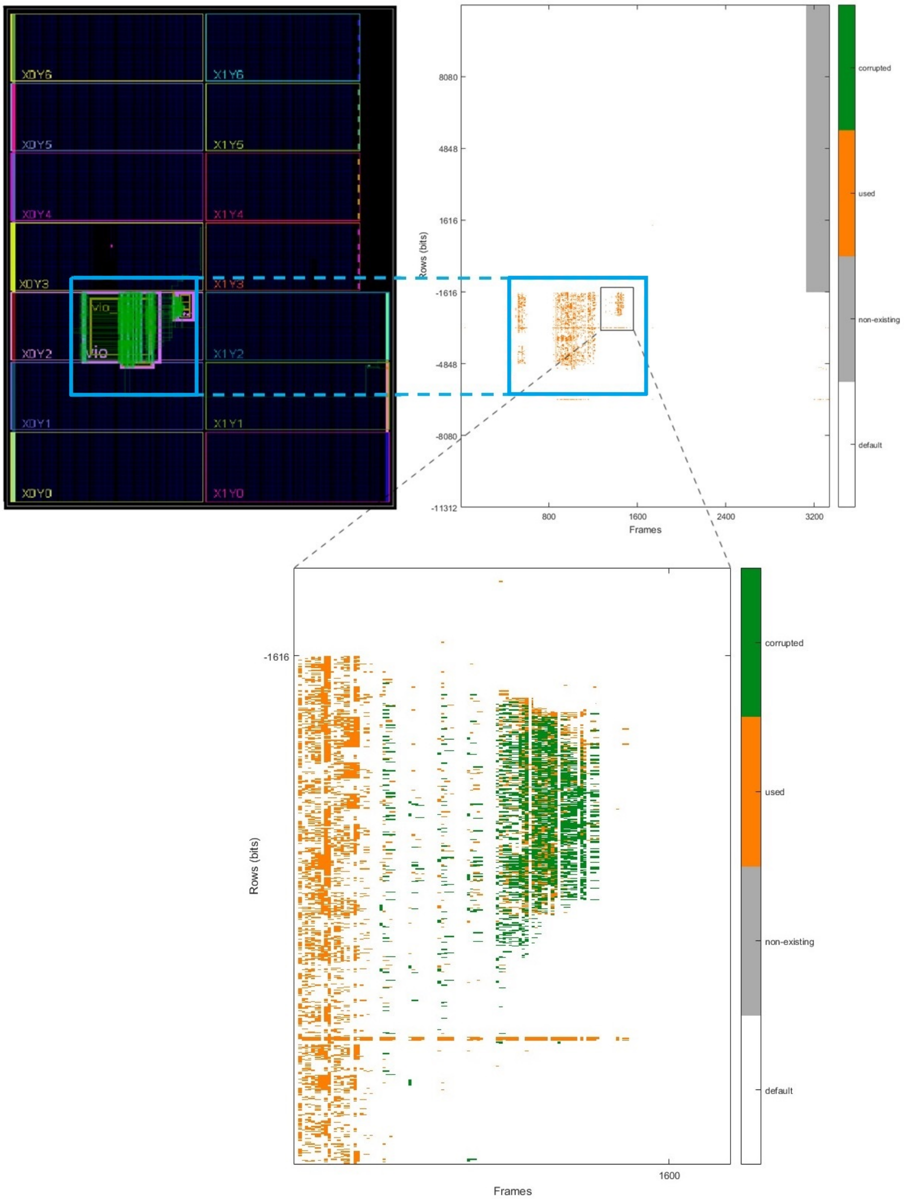


Figure 5.3: Layout of the plain counters implementation, the configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test.

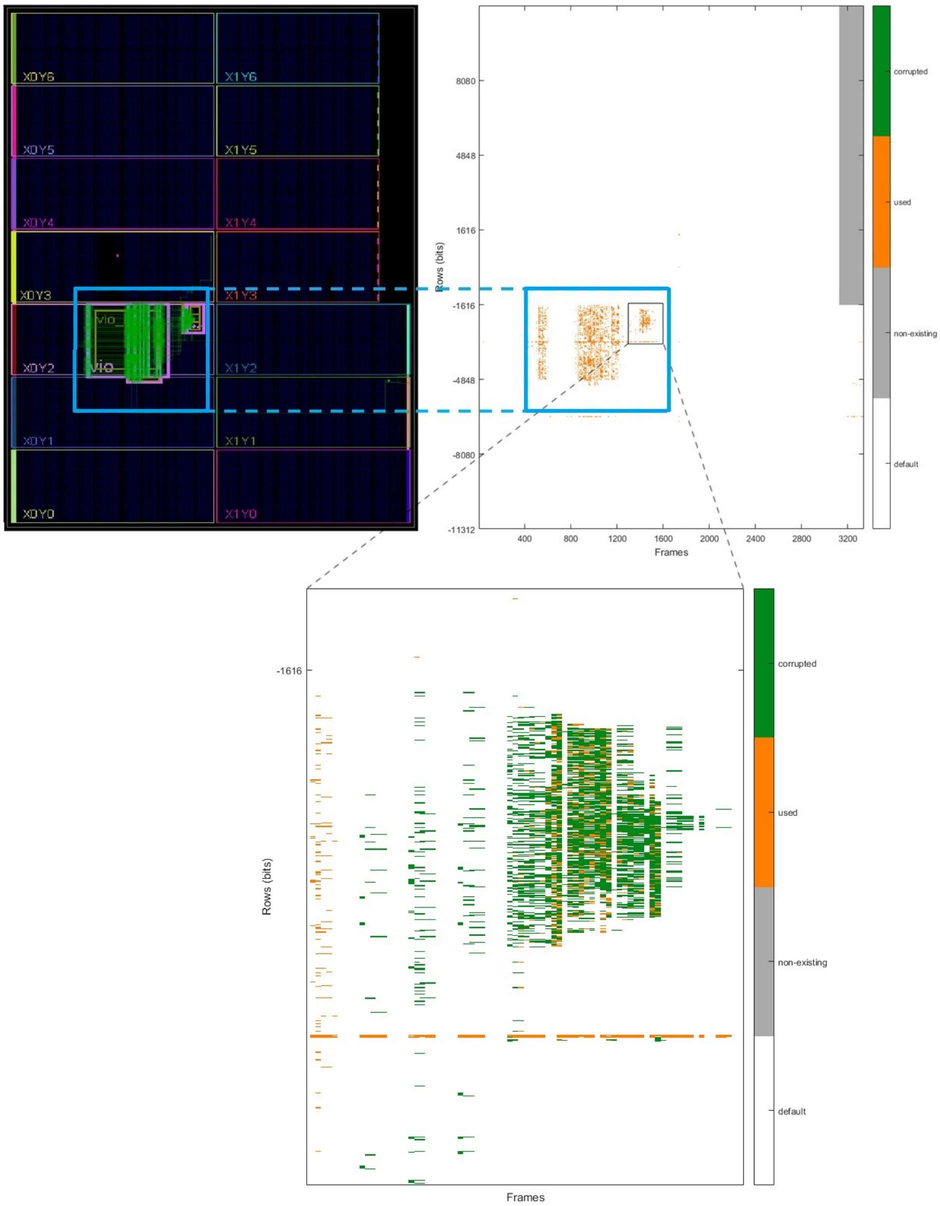


Figure 5.4: Layout of the plain counters implementation with global signals tripled, the configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test.

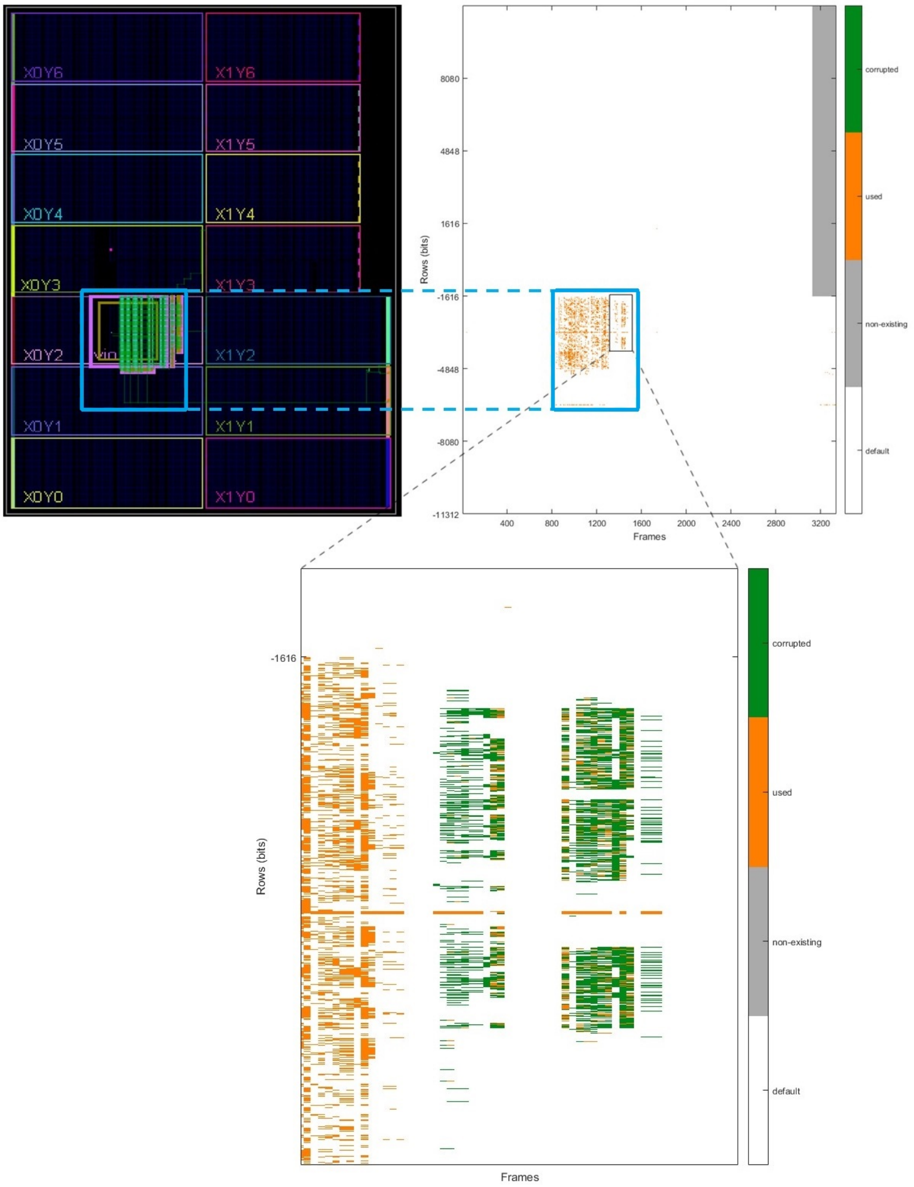


Figure 5.5: Layout of the IDF counters implementation, the configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test.

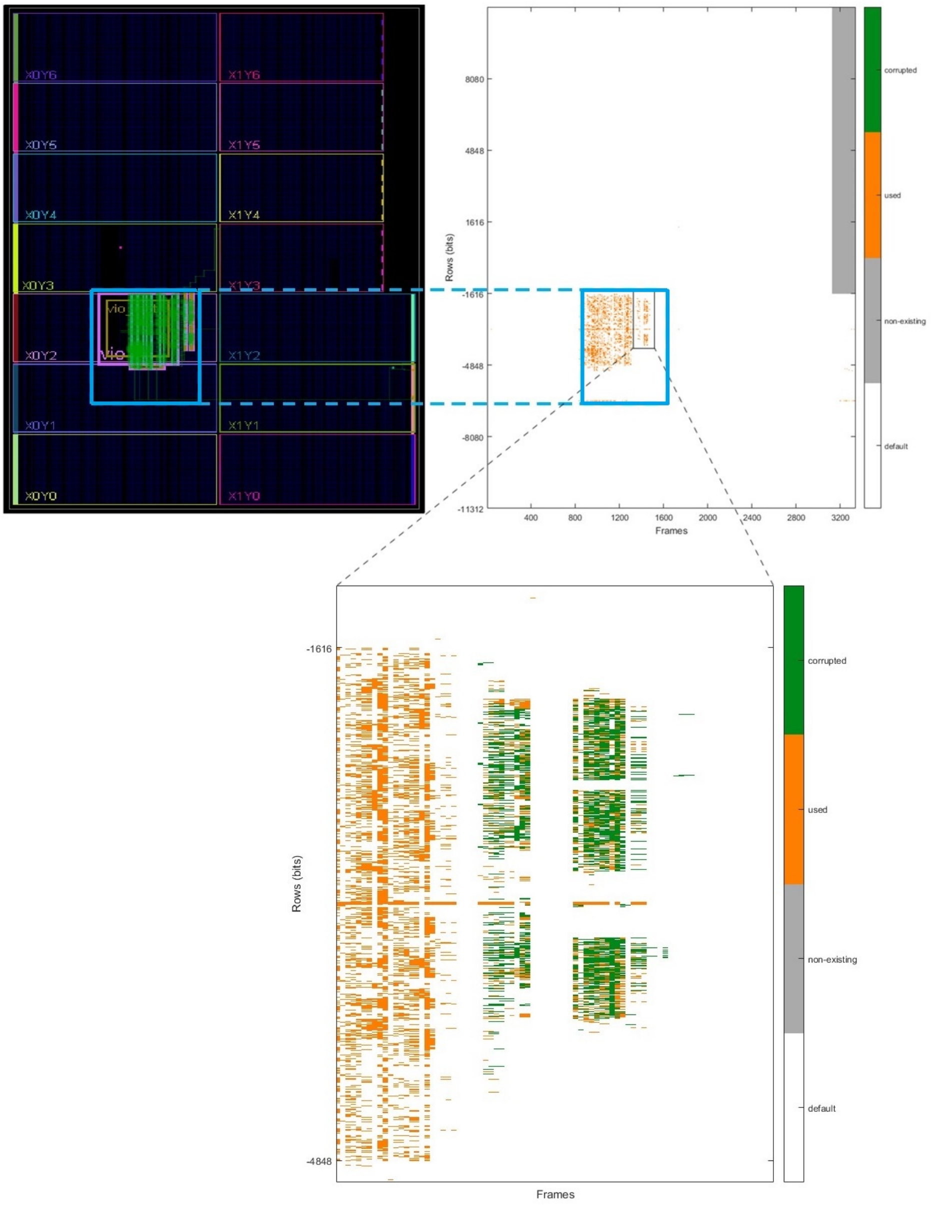


Figure 5.6: Layout of the IDF counters implementation with global signals tripled, the configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test.

Table 5.1 shows the tests results for two implementation without the global signals triplication.

Table 5.1: Test results for the fault injection test on plain and IDF implemented the counters without the triplication of the global signals.

| | **Plain implementation** | **IDF implementation** |
|---|---|---|
| Essential bits | 29957 | 28568 |
| Injected essential bits | 7200 | 7200 |
| Critical bits | 38 | 21 |
| $\mu_{\hat{p}}(\%)$ | 0.52 ± 0.08 | 0.29 ± 0.06 |
| $\epsilon_{\hat{p}}(\%)$ | 16.18 | 21.79 |

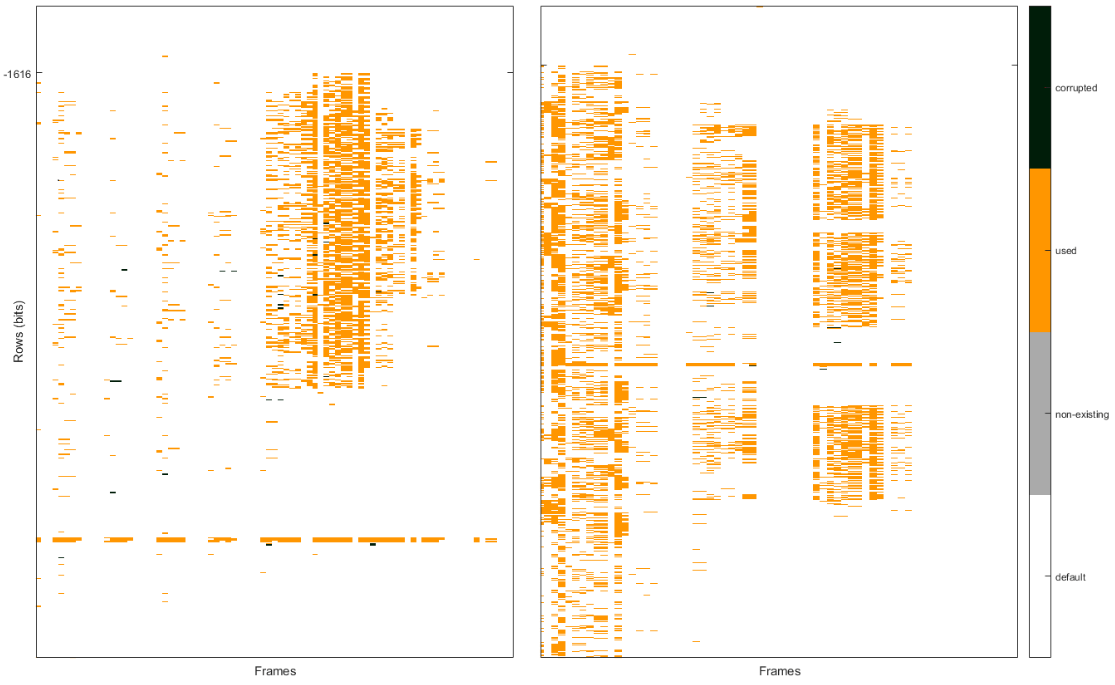
Here, it is possible to deduce that the IDF reduces critical bits by $\simeq$ 45%. Although $\epsilon_{\hat{p}}$ is an high value, the two $\sigma_{\hat{p}}$ obtained from the tests ensure that the results for the estimator do not overlap.

The distribution of the found critical bits for the plain and IDF implementation is represented in Fig. 5.7. In particular, Fig. 5.7(b) shows a presence of critical bits in isolated modules, i.e. a corrupted bit in a counter may cause the failure of the other two. Although each counter operates independently from the others, the corruption of the global logic in a single isolated module may alter the logic of the other modules. Indeed, test results on the two implementation with the triplication of the global logic lead to an absence of critical bits during the fault injection as shown in Table 5.2. Here, the confidence level for the obtained $\hat{p}$ value, is 95%. Indeed the edge of this confidence interval is given by $B(X = 0; n, p) = (1 - p)^n = 0.05$. In particular, given the logarithm of the equality $(1 - p)^n = 0.05$, the value obtained for $\hat{p}$ on the sample reasonably leads to assume a value close to 0 for the population and, thus, to approximate $ln(1 - p)$ by $-p$. Therefore, it turns out that

$$p = -\frac{ln(0.05)}{n} \simeq -\frac{3}{n}, \tag{5.3}$$

that is an equation also known as rule of three. Hence, the rule of three states that if a certain event did not occur in a sample with $n$ trials, the interval from 0 to $3/n$ is a 95% confidence interval for the rate of occurrences in the population.

Although the absence of critical bit the last test may let to suppose that the triplication of the global logic in the C$^3$ can be a rapid and simple way to enhance its correction capability, this solution has limits. Firstly, in the C$^3$, each logic block communicates with the others in a much more complex way compared to the counters. Secondly, the FPGA limitation forbids the triplication of several hardware blocks (e.g. ICAP). Therefore, due to the good IDF impact pointed out from these tests, I have used both IDF and the triplication of the global logic solutions in the second C$^3$ implementation.



(a) (b)

Figure 5.7: Representation of the critical bits in the plain and in the IDF circuits implementation. (a) Map of critical bits in the plain implemented circuit. (b) Map of critical bits in the IDF implemented circuit.

Table 5.2: Test results for the fault injection test on plain and IDF implemented the counters with the triplication of the global signals.

| | **Plain implementation** | **IDF implementation** |
|---|---|---|
| Essential bits | 29589 | 29040 |
| injected essential bits | 7200 | 7200 |
| Critical bits | 0 | 0 |
| $\mu_{\hat{p}}(\%)$ | 0 at 95% C.L. | 0 at 95% C.L. |

## 5.3 Scrubber software and hardware test setup

The tests setup of the C$^3$ fault injection includes the KC705 board interfaced to a computer via UART-over-JTAG. For the test execution, the software part is managed by TCL scripts, interfacing to the developed hardware. The main steps of the test are listed below.

1. **Essential bits generation.** A script elaborates the bitstream and provides the essential bit list for each C$^3$ implementation.

2. **Random sample generation.** A script randomly generates an essential bits subset, providing address and bit offset of the programmed configuration frames used as population sample. The sample size is set by the user.

3. **Firmware upload.** The firmware to be tested is uploaded into the FPGA.

4. **Triplication of the configuration.** A script executes the triplication of the configuration frames.

5. **Injection.** The main test step is executed by means of a script interfaced to the circuit via UART-over-JTAG. Here, the scrubber initialization occurs, i.e. the script sets the number of redundant and empty frames to be scanned, the number of scan cycles before a reset and the Unixtime counter. Then, the script sequentially sets the O and V commands for each configuration bit provided by the random sample generation step. The O command takes and corrupts a bit, whereas the V command corrects it by voting the configuration frame.

If the circuit gives an unexpected response in the injection procedure, the procedure stops after 10 attempted interactions with the hardware. Finally, the test restarts from the firmware upload step each time that a critical bit is found. At each test run, the injected bits are removed from the list in order to have a sample without repetitions. In Fig. 5.8, the main elements of the test execution are depicted.

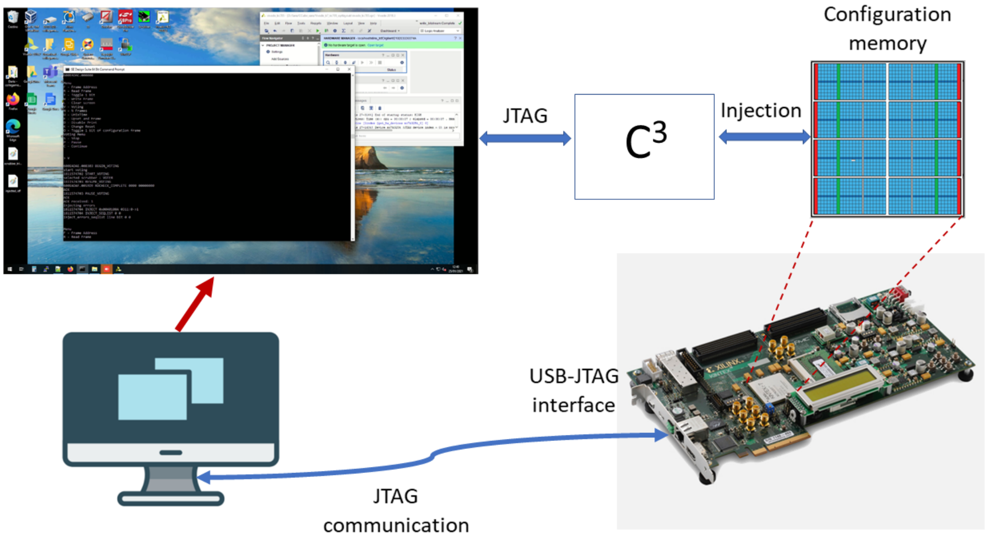


Figure 5.8: Representation of the test set up for the fault-injection tests. The PC interface via JTAG to the FPGA where the configuration memory is corrupted through the injection.

As a remarkable result, the executed test allows to use the same circuit for fault-injection and test of self -corrupting and -repairing capability of the circuits.

## 5.4 C$^3$ tests results

The statistical analysis for the two C$^3$ implementations has been performed by applying the same considerations presented for the test circuits. The sample sizes used to test the scrubbers have been chosen in such a way to obtain $\epsilon_{\hat{p}} \lesssim 10\%$ on the $\mu_{\hat{p}}$ estimator, since the C$^3$ fault-injection tests include the user intervention for each run that requires a suitable additional time. The bit maps for the two implementation are represented in in Fig. 5.9 and in Fig. 5.10 for the plain and IDF implementations, respectively. Since the bit cross-section for irradiation test with 62 MeV protons is known for the Kintex-7 devices [40], it is interesting to derive the equivalent fluence $\Phi_{eq,62MeV}$ (see Sec. 2.1 for the fluence definition), under which the C$^3$ have been subjected during the tests. In the fault injection tests the test target is represented by the circuit essential bits, whereas the target is represented by the entire FPGA during the irradiation. Therefore, the required parameters to evaluate $\Phi_{eq,62MeV}$ are the device cross-section $\sigma_{7K325T,62MeV}$ and the number of hit bits assuming to hit the entire device $N_{inj,eq}$. The former can be calculated from the bit cross-section $sigma_{Kintex-7,62MeV}$ and the Kintex-7 325T total bits (excluding BRAMs) $N_{7K325T}$:

$$\sigma_{7K325T,62MeV} = N_{7K325T} \times \sigma_{Kintex-7,62MeV}. \tag{5.4}$$

Instead, the latter can be calculated by multiplying the number of corrupted bits in the fault injection test $N_{inj}$ by the ratio between the programmed circuit bits $N_{ess}$ and the whole FPGA bits $N_{7K325T}$ for the Kintex-7 325T excluding BRAMs:

$$N_{inj,eq} = N_{inj} \frac{N_{7K325T}}{N_{ess}}. \tag{5.5}$$

Similarly, the number of critical bits for the plain and the IDF C$^3$ essential bits population can be evaluated through the relation

$$N_{critical} = N_{critical,sample} \frac{N_{ess}}{N_{inj}}. \tag{5.6}$$

Since the bit cross-section uncertainty is not provided, I have chosen the less significant digit as such. In the following, the critical bits map for the plain and the IDF C$^3$ are depicted in Fig. 5.11 and 5.11, respectively.

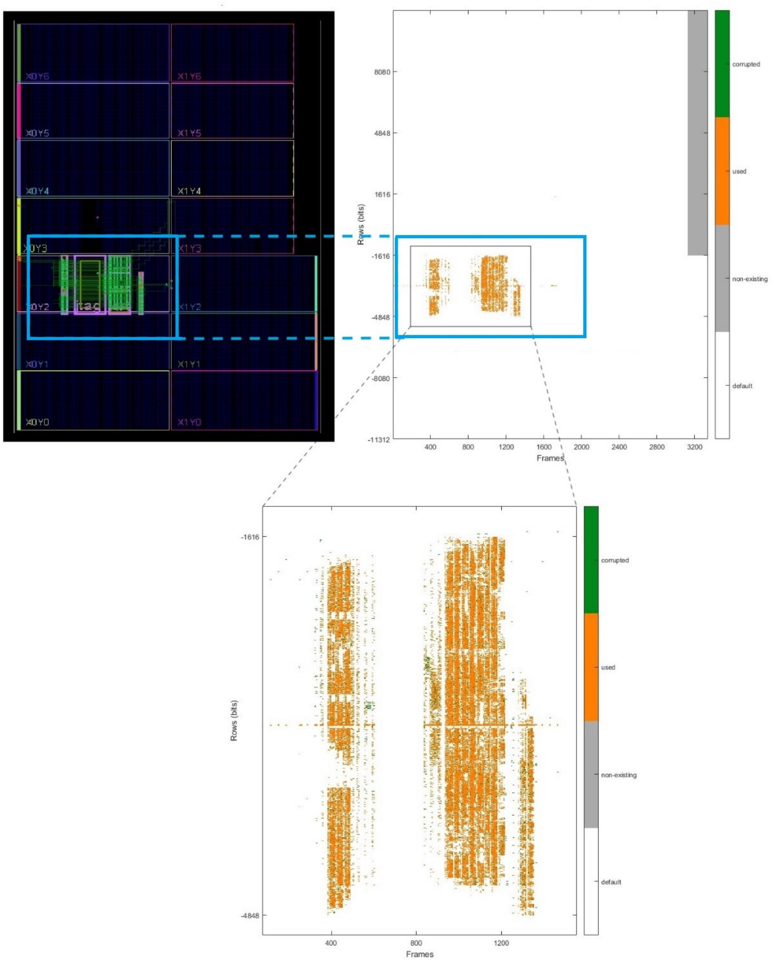


Figure 5.9: Layout of the plain C$^3$ implementation, the pertaining configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test

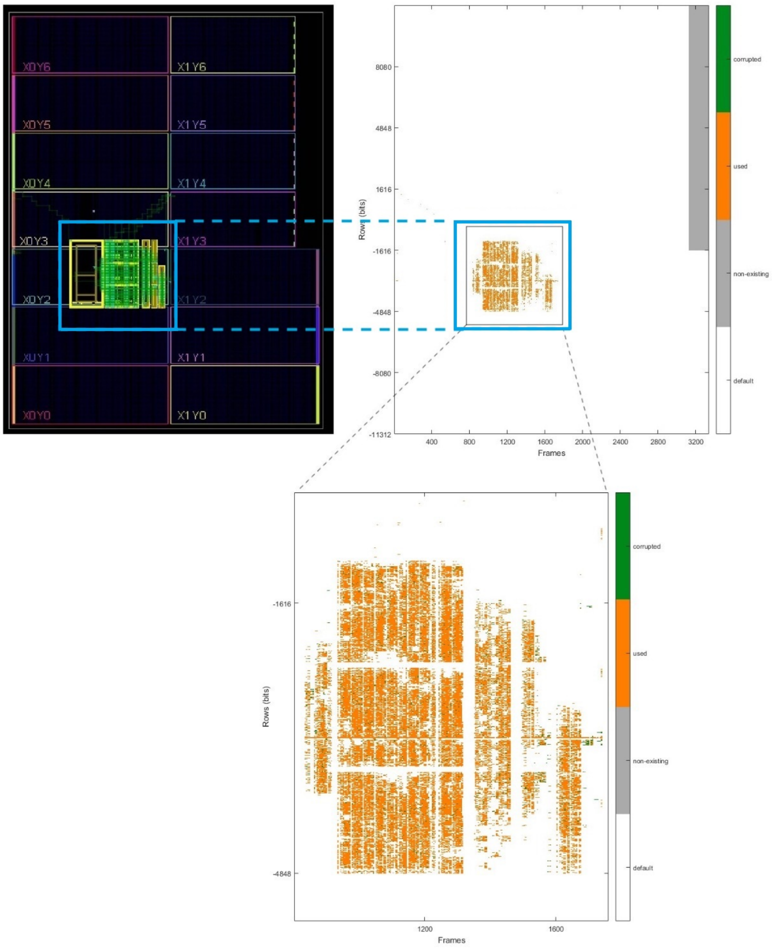


Figure 5.10: Layout of the IDF C$^3$ implementation, the pertaining configuration bit map (orange colour), and the injected essential bits sample in (light-green) in ConfigView. Top left: the implementation view in Vivado. Top right: bit map of the circuit. Bottom: close-up of the bit map representing the essential bits injected during the test.

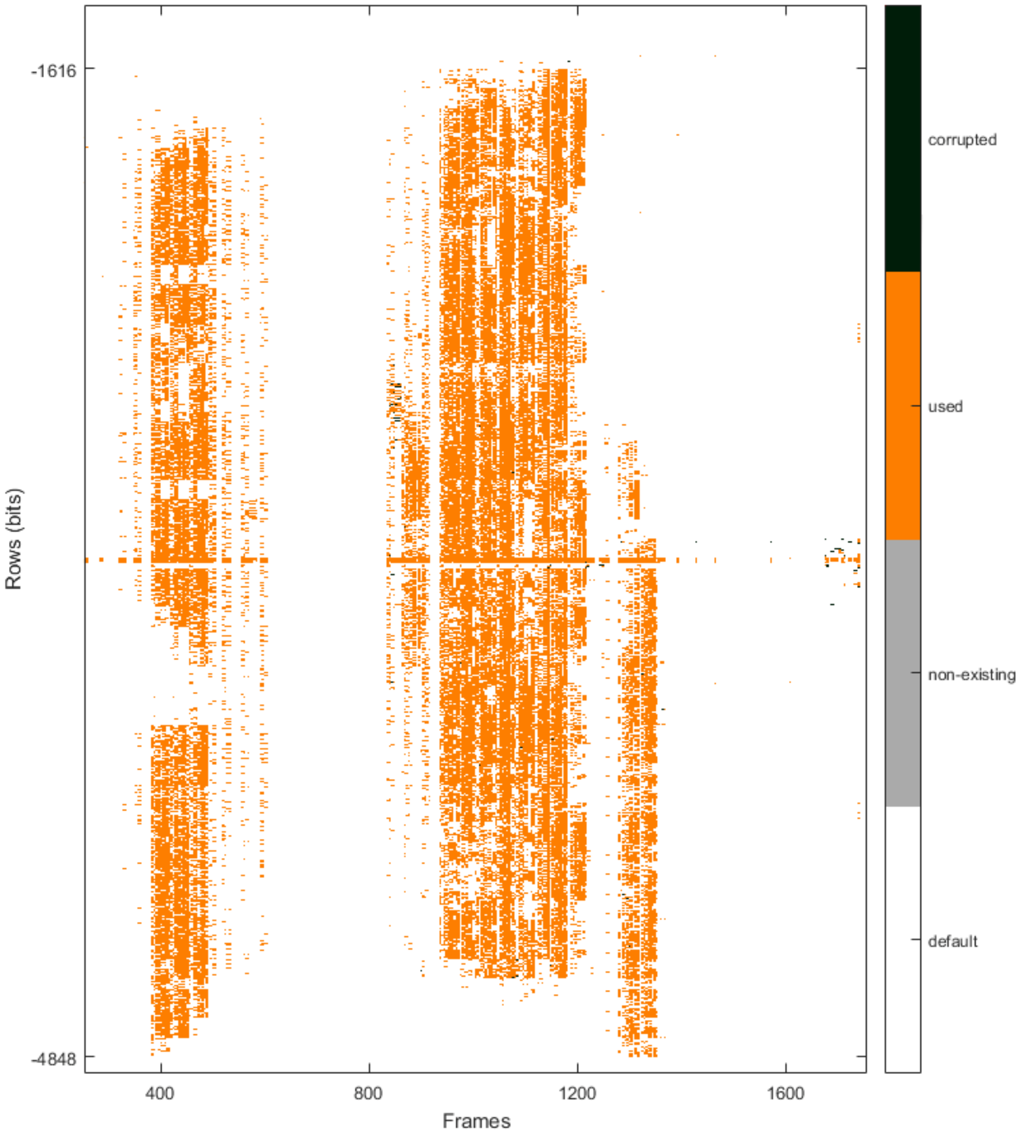


Figure 5.11: Map of the critical bits (dark blue) the C$^3$ plain implementation.

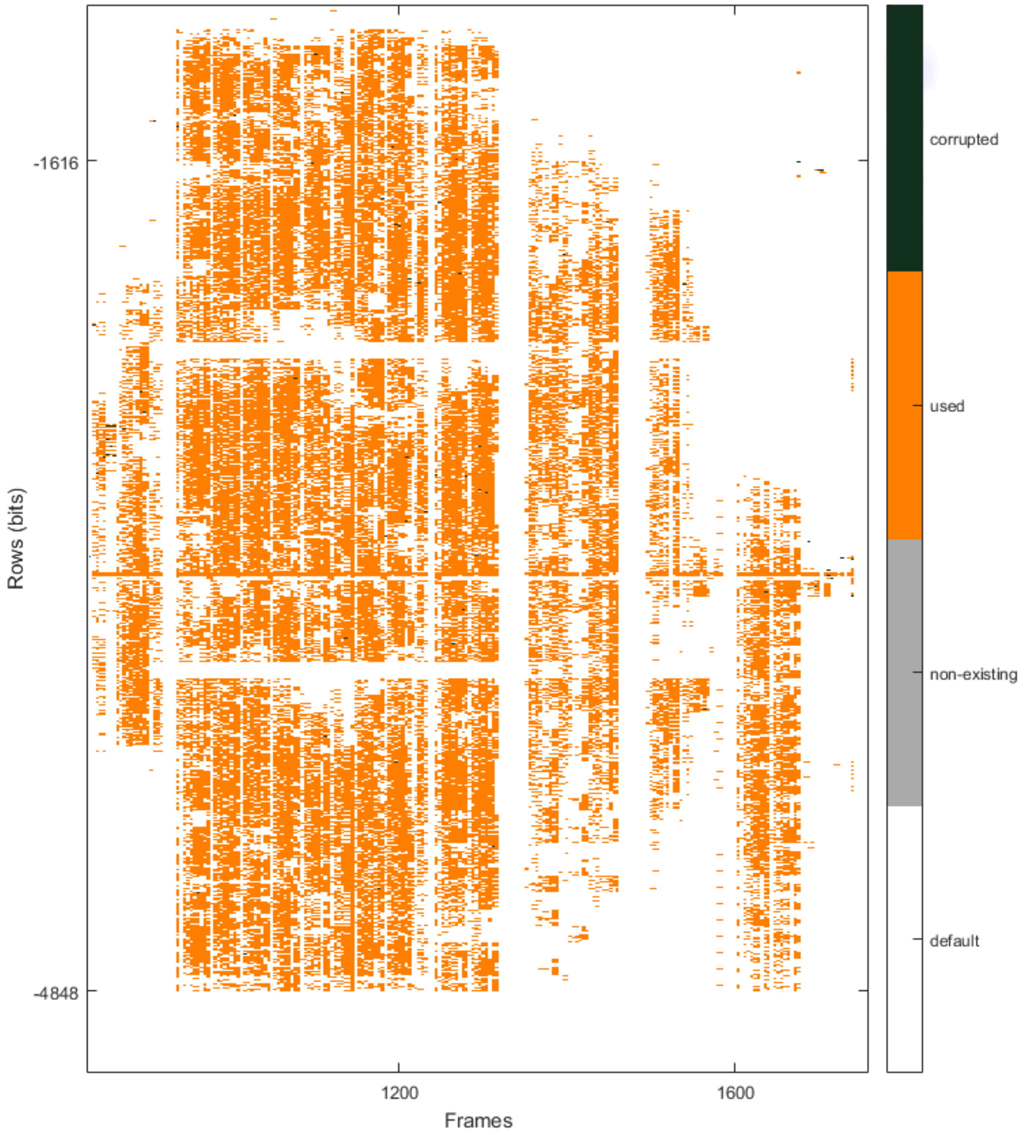
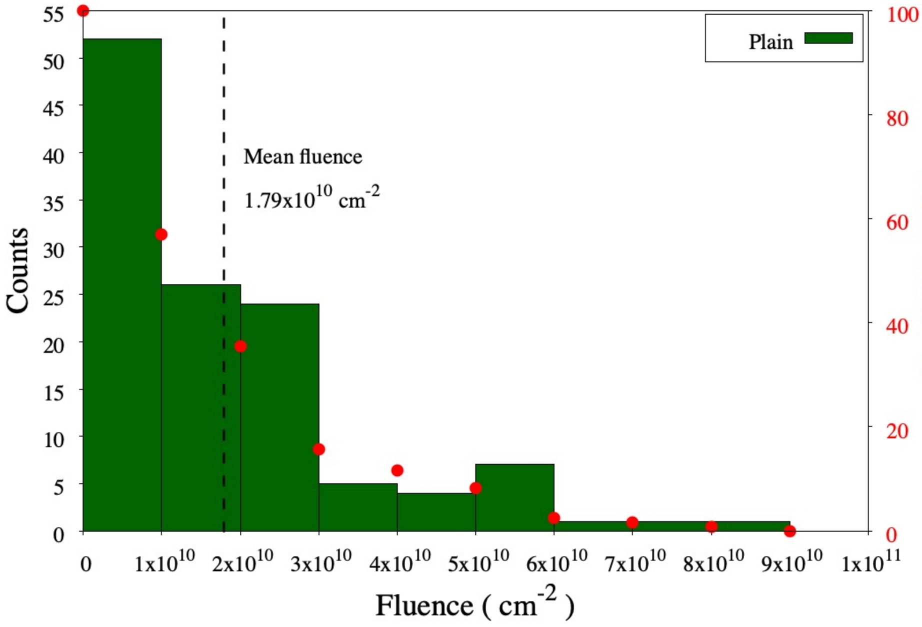


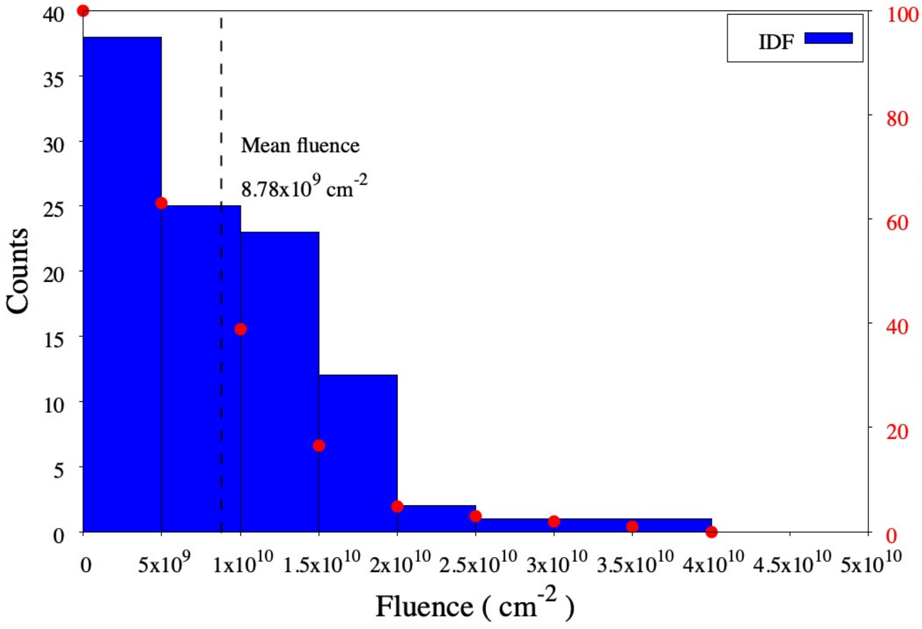
Figure 5.12: Map of critical bits (dark blue) for the IDF scrubber implementation.

In Fig. 5.13, the counts for the equivalent fluences between two failures (two critical bits) is shown for both plain and IDF C$^3$ implementations. The vertical dashed line indicates the mean fluence before a failure, whereas each red point represents the value of the reliability estimated up to a certain value of fluence.

Finally, the test results are presented in the Table 5.3.



(a)



(b)

Figure 5.13: Histogram of the equivalent fluence absorbed before failure and reliability for the two C$^3$ implementations. Note that the two histograms have different horizontal scales. (a) Plain implementation. (b) IDF implementation.

Table 5.3: Test results for the fault injection tests on plain and IDF C$^3$ implementations.

| | **Plain** | **IDF** |
|---|---|---|
| $N_{7K325T}$ (bit) | 72823424 | 72823424 |
| $N_{ess}$ (bit) | 544268 | 731603 |
| $N_{inj}$ (bit) | 5394 | 3000 |
| $\sigma_{C^3,62MeV}$ (cm$^2$) | $(5.6 \pm 0.5)\times10^{-11}$ | $(1.1 \pm 0.1)\times10^{-10}$ |
| $\sigma_{7K325T,62MeV}$ (cm$^2$) | $(3.3 \pm 0.1)\times10^{-7}$ | $(3.3 \pm 0.1)\times10^{-7}$ |
| $\Phi_{eq,62MeV}$ (cm$^{-2}$) | $(2.154 \pm 0.065)\times10^{12}$ | $(8.91 \pm 0.27)\times10^{11}$ |
| $N_{critical,sample}$ (bit) | $(1.2 \pm 0.1)\times10^{2}$ | $(1 \pm 0.1)\times10^{2}$ |
| $N_{critical}$ (bit) | $(1.2 \pm 0.1)\times10^{4}$ | $(2.5 \pm 0.2)\times10^{4}$ |
| $\mu_{\hat{p}}$(%) | 2.2 ± 0.2 | 3.4 ± 0.3 |
| $\epsilon_{\hat{p}}$ (%) | 8.9 | 9.6 |
| Repairable device bits(%) | 99.98 | 99.96 |

The results show that the modifications applied for the second scrubber implementation do not lead to an increase of the auto-correction capability, against what expected from the counters circuits tests results.

The essential bits of the two C$^3$ implementations are of the same order of magnitude, but the sample of the plain implementation is almost double the IDF one. Therefore, although the critical bits are equal within the errors, the equivalent critical bits of the IDF are almost double the plain ones. This means that the IDF implementations provides a lower reliability compared to the plain implementation. Fig. 5.14 reports the histogram representing main failure causes and the relative frequency for the two C$^3$ implementations. Here, I have split the failure causes in four main subcategories described below.

1. **ICAP.** This category includes the failures occurred overcoming of the fixed threshold on the number of errors in a frame through the check described in 4.3. When the ICAP fails, the scrubber reads all the bits set at '1', whereas a correct frame normally includes a few number of bits set at '1'. Here, I have chosen a check threshold of 127 errors per single frame.

2. **UART RX and writing failure.** I have observed C$^3$ failures in the writing frame operation during the injection, followed by the failure of the UART RX.

3. **UART TX.** This category indicates the failures of the UART TX communication.

4. **IDCODE.** The circuit readback an incorrect device IDCODE.

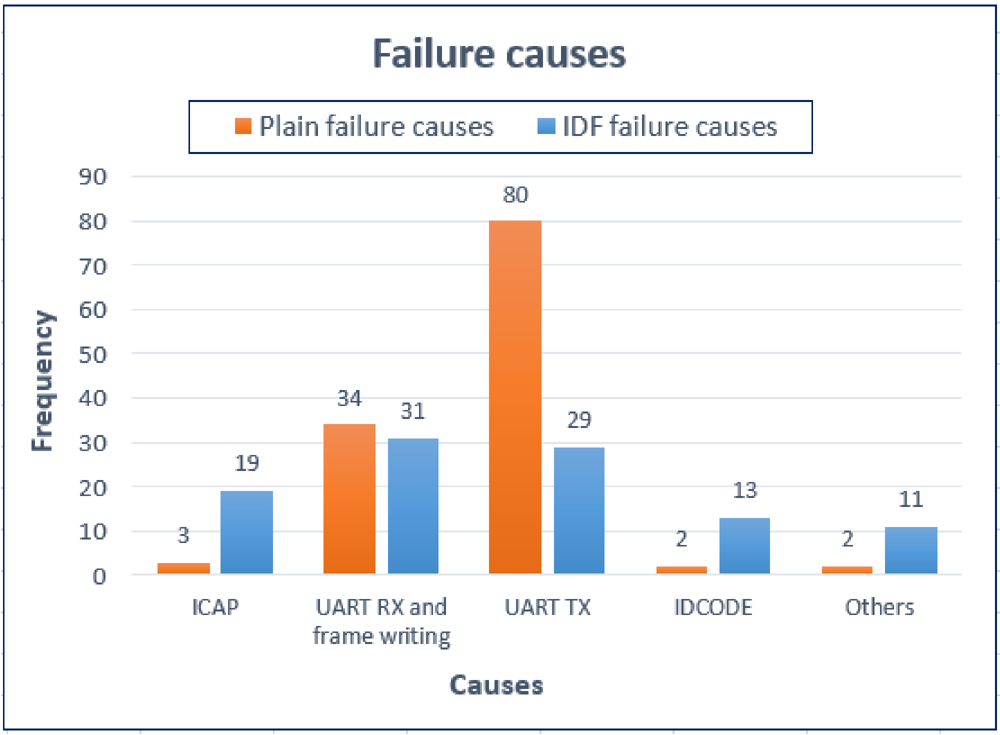


Figure 5.14: Representation of the histogram obtained analysing the plain and IDF $C^3$ failure causes.

Moreover, in Fig. 5.12, it is possible to observe the presence of critical bits localized in the single core, i.e. a bit corruption in a single core may cause the failure of the other two. In the next section, the possible causes of the observed results will be presented.

## 5.5 Lessons learned

The IDF implementation of the $C^3$ needed a modification on the circuit architecture in order to respect the isolation requirements. Indeed, as mentioned in 4.6, I have included Data RAM and Program ROM scrubbers in each core and I have added the Mixer to the plain design. In a TMR-based circuit, the smaller the size of the circuit sub-block to vote, the greater the number of majority voters, i.e. the reduction of the size of the sub-block to vote, increases the TMR granularity. This approach increases probability of masking an error, since the error propagation is limited by the voters operation, providing an increase in the circuit reliability. The scrubber architecture used in the plain $C^3$ implementation is based on a granular TMR system, since the constituting sub-block of each copy has a majority voter at its output. To realize the IDF implementation, I have decreased the granularity level by including Data RAM and Program ROM scrubbers in the relative core. This modification removes the majority voting for the memories scrubbers outputs nd therefore, the error masking on these signals. Moreover, the Mixer may include logic paths merging two or more TMR modules. In particular, differently from the test circuits, the signals passing through the Mixer for the voting operation are fed back into the cores. Hence, the applied changes might reduce reliability explaining:

1. the higher number of the critical bits in the circuit compared to the plain $C^3$ implementation;
2. the fraction of critical bits in a single core is higher in the IDF implementation with respect to the plain one.

For this $C^3$ implementation, not only I have renounced a part of granularity by including the memories scrubbers into the cores, but also the Mixer increases the merging of the signals of different TMR domains.

To avoid the Mixer merging effects, a decrease of the TMR granularity level should be considered, since a lower number of signals to be voted may reduce the merging effect. On the other hand, the merging of an high number of Mixers in the circuit could be mitigated by maintaining the granularity of the plain implementation.

Although it is not obvious what the best alternative is, it is certain that, in general, the application of IDF into a circuit needs a dedicated architecture, i.e. the circuit architecture must be designed from scratch to support IDF.

# Conclusions

This thesis work focused on the design of FPGA-based self-repairing circuits for the FPGA monitoring system of BEAST II, at the Belle II detector. I have proposed two novel designs for the mitigation of the SEUs effects in the FPGA configuration memory, named as plain and IDF C$^3$. I have implemented the C$^3$ architecture for the XC7K325T FPGA of BEAST II and I have modified the previous plain implementation by conceiving a calibration system to find the DCO frequency which allows to easily determine its parameters for any circuit layout. Moreover, my software and hardware changes have improved the C$^3$ plain implementation, since the debugging operation does not need to regenerate the firmware from scratch and the user-circuit interface is significantly simplified. Also, the software reorganization enhanced the circuit versatility for the other BEAST II FPGAs. As far as the IDF implementation is concerned, I have used all the advances and the results obtained through the final plain implementation to fulfil the IDF requirements. Hence, aiming to evaluate the IDF impact on a circuit reliability, I have conceived test circuits, thus performing a statistical analysis of the obtained data. The results demonstrated a decrease in the number of critical bits by using the IDF implementation. Moreover, I have also tripled global signals, that has provided an absence of critical bits in the test circuits. Finally, I have performed dedicated tests on the two C$^3$ implementations, which used the circuit self-corruption capability and aimed to verify the circuits self-repairing performance. Statistical analysis of the obtained results showed that the IDF usage did not increase the self-repair capability, rather decreasing it. Although the essential bits of the two C$^3$ implementations are of the same order of magnitude, tests results demonstrated an increase in the number of critical bits in the IDF implementation with respect to the plain one since for the former $N_{critical} = (2.5 \pm 0.2) \times 10^4$, whereas for the latter $N_{critical} = (1.2 \pm 0.1)\times10^4$ . Indeed, the tests show that the plain C$^3$ implementation allows to repair the 99.98% of the FPGA bits (excluding BRAMS), whereas the IDF provides a lower value of 99.96%. Moreover I have calculated the circuit cross section and the mean fluence before a failure. Regarding the circuit cross section, the IDF one is greater than the plain ones, since their values are $(1.1 \pm 0.1)\times10^{-10}$ cm$^2$ and $(5.6 \pm 0.5)\times10^{-11}$ cm$^2$, respectively, i.e. the probability to observe a failure of the IDF implementation is almost twice the plain one. As far as the mean fluence between failure, the values obtained from the test results show that the plain implementation reach an higher self-repairing capability, since for the IDF is $8.78\times10^9$ cm$^{-2}$, whereas for the plain implementation it is $1.79\times10^{10}$ cm$^{-2}$. Finally, the test results demonstrate that the IDF C$^3$ implementation does not improve the error correction, rather slightly worsening it. In the final part of the thesis work I have investigated on the possible causes of this issue, studying how the architecture modification may violate the isolation between logic blocks.

For the improvement previously discussed and the test results obtained, the plain C$^3$ firmware is now running on the XC7K325T FPGA of BEAST II. Nevertheless, my work with the IDF C$^3$ implementation provides insight starting point for the development of self-repair circuits based on the Xilinx IDF. this in turns advances the state-of-the-art in the use of the FPGAs in the on-detector electronics of the HEP experiments.

# Bibliography

[1] Wei-Shu Hou, Phys. Rev. D **48**, 2342 (1993).

[2] Lincoln Wolfenstein, Phys. Rev. Lett. **51**, 1945 (1983).

[3] M. Tanabashi, K. Hagiwara, K. Hikasa, K. Nakamura, Y. Sumino, F. Takahashi, J. Tanaka, K. Agashe, G. Aielli, C. Amsler, M. Antonelli, D. M. Asner, H. Baer, Sw. Banerjee, R. M. Barnett, T. Basaglia, C. W. Bauer, J. J. Beatty, V. I. Belousov, J. Beringer, S. Bethke, A. Bettini, H. Bichsel, O. Biebel, K. M. Black, E. Blucher, O. Buchmuller, V. Burkert, M. A. Bychkov, R. N. Cahn, M. Carena, A. Ceccucci, A. Cerri, D. Chakraborty, M.-C. Chen, R. S. Chivukula, G. Cowan, O. Dahl, G. D'Ambrosio, T. Damour, D. de Florian, A. de Gouvêa, T. DeGrand, P. de Jong, G. Dissertori, B. A. Dobrescu, M. D'Onofrio, M. Doser, M. Drees, H. K. Dreiner, D. A. Dwyer, P. Eerola, S. Eidelman, J. Ellis, J. Erler, V. V. Ezhela, W. Fetscher, B. D. Fields, R. Firestone, B. Foster, A. Freitas, H. Gallagher, L. Garren, H.-J. Gerber, G. Gerbier, T. Gershon, Y. Gershtein, T. Gherghetta, A. A. Godizov, M. Goodman, C. Grab, A. V. Gritsan, C. Grojean, D. E. Groom, M. Grünewald, A. Gurtu, T. Gutsche, H. E. Haber, C. Hanhart, S. Hashimoto, Y. Hayato, K. G. Hayes, A. Hebecker, S. Heinemeyer, B. Heltsley, J. J. Hernández-Rey, J. Hisano, A. Höcker, J. Holder, A. Holtkamp, T. Hyodo, K. D. Irwin, K. F. Johnson, M. Kado, M. Karliner, U. F. Katz, S. R. Klein, E. Klempt, R. V. Kowalewski, F. Krauss, M. Kreps, B. Krusche, Yu. V. Kuyanov, Y. Kwon, O. Lahav, J. Laiho, J. Lesgourgues, A. Liddle, Z. Ligeti, C.-J. Lin, C. Lippmann, T. M. Liss, L. Littenberg, K. S. Lugovsky, S. B. Lugovsky, A. Lusiani, Y. Makida, F. Maltoni, T. Mannel, A. V. Manohar, W. J. Marciano, A. D. Martin, A. Masoni, J. Matthews, U.-G. Meißner, D. Milstead, R. E. Mitchell, K. Mönig, P. Molaro, F. Moort-

gat, M. Moskovic, H. Murayama, M. Narain, P. Nason, S. Navas, M. Neubert, P. Nevski, Y. Nir, K. A. Olive, S. Pagan Griso, J. Parsons, C. Patrignani, J. A. Peacock, M. Pennington, S. T. Petcov, V. A. Petrov, E. Pianori, A. Piepke, A. Pomarol, A. Quadt, J. Rademacker, G. Raffelt, B. N. Ratcliff, P. Richardson, A. Ringwald, S. Roesler, S. Rolli, A. Romaniouk, L. J. Rosenberg, J. L. Rosner, G. Rybka, R. A. Ryutin, C. T. Sachrajda, Y. Sakai, G. P. Salam, S. Sarkar, F. Sauli, O. Schneider, K. Scholberg, A. J. Schwartz, D. Scott, V. Sharma, S. R. Sharpe, T. Shutt, M. Silari, T. Sjöstrand, P. Skands, T. Skwarnicki, J. G. Smith, G. F. Smoot, S. Spanier, H. Spieler, C. Spiering, A. Stahl, S. L. Stone, T. Sumiyoshi, M. J. Syphers, K. Terashi, J. Terning, U. Thoma, R. S. Thorne, L. Tiator, M. Titov, N. P. Tkachenko, N. A. Törnqvist, D. R. Tovey, G. Valencia, R. Van de Water, N. Varelas, G. Venanzoni, L. Verde, M. G. Vincter, P. Vogel, A. Vogt, S. P. Wakely, W. Walkowiak, C. W. Walter, D. Wands, D. R. Ward, M. O. Wascko, G. Weiglein, D. H. Weinberg, E. J. Weinberg, M. White, L. R. Wiencke, S. Willocq, C. G. Wohl, J. Womersley, C. L. Woody, R. L. Workman, W.-M. Yao, G. P. Zeller, O. V. Zenin, R.-Y. Zhu, S.-L. Zhu, F. Zimmermann, P. A. Zyla, J. Anderson, L. Fuller, V. S. Lugovsky, and P. Schaffner, Phys. Rev. D **98**, 030001 (2018).

[4] Yukiyoshi Ohnishi, Tetsuo Abe, Toshikazu Adachi, Kazunori Akai, Yasushi Arimoto, Kiyokazu Ebihara, Kazumi Egawa, John Flanagan, Hitoshi Fukuma, Yoshihiro Funakoshi, Kazuro Furukawa, Takaaki Furuya, Naoko Iida, Hiromi Iinuma, Hoitomi Ikeda, Takuya Ishibashi, Masako Iwasaki, Tatsuya Kageyama, Susumu Kamada, Takuya Kamitani, Ken-ichi Kanazawa, Mitsuo Kikuchi, Haruyo Koiso, Mika Masuzawa, Toshihiro Mimashi, Takako Miura, Takashi Mori, Akio Morita, Tatsuro Nakamura, Kota Nakanishi, Hiroyuki Nakayama, Michiru Nishiwaki, Yujiro Ogawa, Kazuhito Ohmi, Norihito Ohuchi, Katsunobu Oide, Toshiyuki Oki, Masaaki Ono, Masanori Satoh, Kyo Shibata, Masaaki Suetake, Yusuke Suetsugu, Ryuhei Sugahara, Hiroshi Sugimoto, Tsuyoshi Suwada, Masafumi Tawada, Makoto Tobiyama, Noboru Tokuda, Kiyosumi Tsuchiya, Hiroshi Yamaoka, Yoshiharu Yano, Mitsuhiro Yoshida, Shin-ichi Yoshimoto, Demin Zhou, and Zhanguo Zong, Progress of Theoretical and Experimental Physics **2013**, (2013), 03A011.

[5] Marko Bračko and the SuperBelle Collaboration, Journal of Physics: Conference Series **171**, 012098 (2009).

[6] Jake Bennett, International Journal of Modern Physics: Conference Series **46**, 1860082 (2018).

[7] Mitsuo Kikuchi, Tetsuo Abe, Kazumi Egawa, Hitoshi Fukuma, Kazuro Furukawa, Naoko Iida, Hitomi Ikeda, Takuya Kamitani, Ken-Ichi Kanazawa, K. Ohmi, Katsunobu Oide, Kyo Shibata, Masafumi Tawada, Makoto Tobiyama, and Demin Zhou, IPAC 2010 - 1st International Particle Accelerator Conference (2011).

[8] A. Piwinski, The Touschek Effect in Strong Focusing Storage Rings, 1999.

[9] H. Tanigawa, K. Adamczyk, H. Aihara, T. Aziz, S. Bacher, S. Bahinipati, G. Batignani, J. Baudot, P.K. Behera, S. Bettarini, T. Bilka, A. Bozek, F. Buchsteiner, G. Casarosa, D. Červenkov, Y.Q. Chen, L. Corona, T. Czank, S.B. Das, N. Dash, G. de Marino, Z. Doležal, G. Dujany, F. Forti, M. Friedl, E. Ganiev, B. Gobbo, S. Halder, K. Hara, S. Hazra, T. Higuchi, C. Irmler, A. Ishikawa, H.B. Jeon, C. Joo, M. Kaleta, A.B. Kaliyar, J. Kandra, K.H. Kang, P. Kapusta, P. Kodyš, T. Kohriki, M. Kumar, R. Kumar, P. Kvasnička, C. La Licata, K. Lalwani, L. Lanceri, S.C. Lee, Y.B. Li, J. Libby, T. Lueck, S. Maity, S.N. Mayekar, G.B. Mohanty, J.A. Mora Grimaldo, T. Morii, K.R. Nakamura, H. Nakayama, Z. Natkaniec, Y. Onuki, W. Ostrowicz, A. Paladino, E. Paoloni, H. Park, K.K. Rao, I. Ripp-Baudot, G. Rizzo, N. Rout, D. Sahoo, L. Santelj, N. Sato, C. Schwanda, J. Suzuki, S. Tanaka, R. Thalmeier, T. Tsuboyama, Y. Uematsu, S.E. Vahsen, O. Verbycka, L. Vitale, K. Wan, S. Watanuki, J. Webb, J. Wiechczynski, H. Yin, L. Zani, and T. Zhang, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **982**, 164580 (2020).

[10] Andreas Moll, Comprehensive study of the background for the Pixel Vertex Detector at Belle II, 2015.

[11] B. Schwenker et al., PoS **VERTEX2018**, 006 (2019).

[12] C. Marinas, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **731**, 31 (2013), pIXEL 2012.

[13] P.M. Lewis, I. Jaegle, H. Nakayama, A. Aloisio, F. Ameli, M. Barrett, A. Beaulieu, L. Bosisio, P. Branchini, T.E. Browder, A. Budano, G. Cautero, C. Cecchi, Y.-T. Chen, K.-N. Chu, D. Cinabro, P. Cristaudo, S. de Jong, R. de Sangro, G. Finocchiaro, J. Flanagan, Y. Funakoshi, M. Gabriel, R. Giordano, D. Giuressi, M.T. Hedges, N. Honkanen, H. Ikeda, T. Ishibashi, H. Kaji, K. Kanazawa, C. Kiesling, S. Koirala, P. KriÅ¾an, C. La Licata, L. Lanceri, J.-J. Liau, F.-H. Lin, J.-C. Lin, Z. Liptak, S. Longo, E. Manoni, C. Marinas, K. Miyabayashi, E. Mulyani, A. Morita, M. Nakao, M. Nayak, Y. Ohnishi, A. Passeri, P. Poffenberger, M. Ritzert, J.M. Roney, A. Rossi, T. RÃ¶der, R.M. Seddon, I.S. Seong, J.-G. Shiu, F. Simon, Y. Soloviev, Y. Suetsugu, M. Szalay, S. Terui, G. Tortone, S.E. Vahsen, N. van der Kolk, L. Vitale, M.-Z. Wang, H. Windel, and S. Yokoyama, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **914**, 69 (2019).

[14] E. A. Burke, IEEE Transactions on Nuclear Science **33**, 1276 (1986).

[15] H. Spieler, *Semiconductor Detector Systems*, *Series on Semiconductor Science and Technology* (OUP Oxford, ADDRESS, 2005).

[16] M. Huhtinen, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **491**, 194 (2002).

[17] Amy A. Cordones and Stephen R. Leone, Chem. Soc. Rev. **42**, 3209 (2013).

[18] G. L. Tan, M. F. Lemon, D. J. Jones, and R. H. French, Phys. Rev. B **72**, 205117 (2005).

[19] Dariusz Makowski, Ph.D. thesis, Lodz, Tech. U., 2006.

[20] Surendra Singh Rathod, A. K. Saxena, and Sudeb Dasgupta, IETE Technical Review **28**, 451 (2011).

[21] P. E. Dodd and L. W. Massengill, IEEE Transactions on Nuclear Science **50**, 583 (2003).

[22] T.S. Nidhin, Anindya Bhattacharyya, R.P. Behera, T. Jayanthi, and K. Velusamy, Nuclear Engineering and Technology **49**, 1589 (2017).

[23] J. S. Browning, M. P. Connors, C. L. Freshman, and G. A. Finney, IEEE Transactions on Nuclear Science **35**, 1557 (1988).

[24] Rémi Gaillard, in *Soft Errors in Modern Electronic Systems*, edited by Michael Nicolaidis (Springer US, Boston, MA, 2011), pp. 27–54.

[25] F. Siegle, Ph.D. thesis, 2016.

[26] Sophie Duzellier, Aerospace Science and Technology **9**, 93 (2005).

[27] Raffaele Giordano, Barbieri, Perrella, and Roberto Catalano, Instruments **3**, 56 (2019).

[28] A. Aloisio, V. Bocci, R. Giordano, V. Izzo, L. Sterpone, and M. Violante, IEEE Transactions on Nuclear Science **60**, 3502 (2013).

[29] L. Sterpone, *Electronics System Design Techniques for Safety Critical Applications*, *Lecture Notes in Electrical Engineering* (Springer Netherlands, ADDRESS, 2008).

[30] M. Bellato, P. Bernardi, D. Bortolato, A. Candelori, M. Ceschia, A. Paccagnella, M. Rebaudengo, M. S. Reorda, M. Violante, and P. Zambolin, in *Proceedings Design, Automation and Test in Europe Conference and Exhibition* (PUBLISHER, ADDRESS, 2004), Vol. 1, pp. 584–589 Vol.1.

[31] *7 Series FPGAs Configuration, User Guide UG470, v1.13.1* (Xilinx Inc., San Jose, CA, USA, 2018).

[32] Robert Le, Soft Error Mitigation Using Prioritized Essential Bits, 2012.

[33] A. Aloisio, F. Ameli, A. Anastasio, P. Branchini, F. Di Capua, R. Giordano, V. Izzo, and G. Tortone, IEEE Transactions on Nuclear Science **64**, 1185 (2017).

[34] B. G. Taylor, IEEE Transactions on Nuclear Science **45**, 821 (1998).

[35] V. Bocci, G. Chiodi, F. Iacoangeli, R. Nobrega, D. Pinci, and W. Rinaldi, in *IEEE Nuclear Science Symposium Conference Record, 2005* (PUBLISHER, ADDRESS, 2005), Vol. 1, pp. 398–402.

[36] R. Giordano, F. Ameli, P. Bifulco, V. Bocci, S. Cadeddu, V. Izzo, A. Lai, S. Mastroianni, and A. Aloisio, IEEE Transactions on Nuclear Science **62**, 3163 (2015).

[37] Raffaele Giordano, Sandro Cadeddu, Alberto Aloisio, Fabrizio Ameli, Valerio Bocci, Vincenzo Izzo, Adriano Lai, and Stefano Mastroianni, Digitally Controlled Oscillator (DCO) architecture, Patent WO/2016/071813, 12.05.2016.

[38] *Zynq-7000 AP SoCs or 7 Series FPGAs Isolation Design Flow Lab (Vivado Design Suite)* (Xilinx Inc., San Jose, CA, USA, 2016).

[39] *Statistical quality control* (McGraw-Hill Science/Engineering/Math, New York, NY, USA, 1996), Chap. 20.

[40] Raffaele Giordano, Sabrina Perrella, Vincenzo Izzo, Giuliana Milluzzo, and Alberto Aloisio, IEEE Trans. Nucl. Sci. **64**, 2497 (2017).