Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Elettronica

Master Thesis

Temperature Effects on Soft Error Rate Due to Atmospheric Neutrons on 28 nm FPGAs

Advisor: Alessandro Paccagnella
Coadvisor: Paolo Rech

Student: Giovanni Bruni

This work is the outcome of a collaboration with
Universidade Federal do Rio Grande do Sul

UFRGS
UNIVERSIDADE FEDERAL
DO RIO GRANDE DO SUL

15/04/2014

This document has been written using LaTeX on Debian GNU/Linux.

This is an ex parrot.

Monty Python

Contents

1. Introduction

Reliability has an important role in electronic design. A device or a circuit has to guarantee a good level of performances during its lifetime, so a prediction of the ageing and the behaviour taken in particular cases is indispensable.

If we are talking about avionics, aerospace, automotive or particles accelerators, the reliability is a more important concern than in other fields, since human lives and large capital amounts are involved. Many standards, such as *ISO 26262* for automotive or *EUROCAE ED-12B* for avionics or *EN 9100* for aerospace applications, have been developed to achieve and guarantee functional safety.

Many *fault tolerance* techniques useful to avoid problems in electronic devices have been developed. However these techniques are not *free of charge* in an electronic design: these techniques are characterized with *overheads* for area and power consumption. Therefore the designer who uses a particular technique has to find a trade-off among these problems, guaranteeing a good level of reliability without increasing too much the power consumption and the occupation of the electronic device. For achieving a good trade-off the designer needs a good knowledge of the device and of the environment.

Programmable logic devices are particularly appealing, as hardening techniques could be applied at the circuit level without the need of a foundry. Programmable logic is so promising that even the *European Space Agency* (ESA) [1] and the *National Aeronautics and Space Administration* (NASA) [2] themselves suggest the use of commercial components for space designs. The main problem concerning the use of *Commercial Off-The-Shelves* (COTS) FPGA is that they are not directly suitable for radiation environments. The use of hardening techniques is mandatory to ensure the required fault tolerance. As we have described above, during the design process the technique is chosen to guarantee the required SER with the smallest power and area overheads. Therefore an accurate knowledge about the effectiveness of every technique is necessary to avoid error underestimation or the introduction of useless overhead.

In aerospace and avionics the environment is very different from the ground one. As a consequence we need to study the radiation in these environments to understand which interactions can occur with electronic devices.

1.1 Radioactive Environments

There are various fields where rad-hard electronics is employed: particles accelerators, aeroplanes, satellites are some examples. These applications are all characterized by the presence of energetic particles and electromagnetic radiations that are able to create problems in electronic circuits. The main particles and radiations involved in these environments are:

- **Protons:** they are positive particles (charge $+e$) with a mass of about 1.672×10^{-27} kg. The main source is the sun during corona mass ejections. Protons are mainly trapped in Van Allen belts[1], but they can be generated in the atmosphere by cosmic rays[2].

- **Heavy Ions:** they are atoms which have lost one or more electrons, so they present a great positive charge. Their mass is much greater than that of protons: depends on the number of protons and neutrons in the nucleus.

- **Neutrons:** they have no charge. Their mass is around that of protons. The neutron radiation is better explained in Chapter 2.

- **Electrons:** they are negative particles (charge $-e$) and their mass is very small (about 9.109×10^{-31} kg). They come from the sun or they are trapped, as the protons, in Van Allen belts.

- **Muons:** they are created during the reactions cascades provoked by cosmic rays when they reach the atmosphere (see Chapter 2). Their mass is about 1.884×10^{-28} kg, about 200 times the mass of the electron, and they have negative charge ($-e$). They are generated mainly in the upper layers of atmosphere by cosmic rays. Although their mean lifetime is short (2.196×10^{-6} s), thanks to their high speeds and relativistic effects (*time dilation*), they can reach the ground.

[1]These belts are regions around the Earth where protons, electrons and ions are trapped because of the Earth magnetic field.

[2]With *cosmic rays* we mean extremely high energy (up to 10^{11} GeV) and released from the Big Bang or from Super Novas explosions or from the sun during some critical events.

1.2 Radiation-Induced Effects on Electronic Components

The types of radiation listed above induce various effects on the electronic components. When the problem about induced effects by radiation is engaged, there are two basic phenomena to take into account.

The first one is that if a particle has charge, passing through the matter it generates electron-hole pairs (*ionization*) which can then recombine among them or be separated and collected because of the presence of an electric field. The second one concerns the particle mass: if the particle has a big mass, colliding with atoms in a lattice it can move these atoms from their right position, creating interstitial-vacancy pairs (*displacement damage*).

These effects can lead to different types of errors, divided into three main groups:

- **Hard Errors:** the device is irreparably damaged by the radiation event. Some examples: *Single Event Latch-up (SEL)*, *Single Event Gate Rupture (SEGR)*, *Single Event Burnout (SEB)*. In these cases a component is compromised, so the circuit can't work.

- **Soft Errors:** the device is not permanently damaged by the radiation. Usually the output data is corrupted or a not-correct data is stored. This is the case of *Single Event Upset (SEU)*, *Single Event Transient (SET)*, *Single Event Functional Interrupt (SEFI)*.

- **Intermittent Errors:** in this case the device is *out-of-order* when it is exposed to a stated radiation intensity. After the exposure it restarts to work properly. This kind of errors can be a signal of degradation state for the device.

Obviously a soft error can lead to a system failure, for example when the device is used in a down stream process: the incorrect output can reset the system or call a wrong instruction.

In this work we shall analyse only SEUs in the configuration memory of the FPGA (an explanation about this soft error can be found in Section 2.3.1). We ignore SETs in the implemented circuit memory as well as the other possible soft errors. This is done because SETs and other errors are much less frequent than SEUs, so we preferred to focus only on the latter. Furthermore neutrons are more incisive in causing SEUs than SETs and others.

1.3 Field Programmable Gate Array (FPGA)

The Field Programmable Gate Array (FPGA) is an integrated circuit with a regular array architecture where a basic structure is replicated, providing a good level of interconnection among these elementary blocks. This structure includes *Look Up Tables*, that is a circuit useful for implementing logic functions, basic logic blocks, such as adders, multipliers etc, embedded memory (flip-flop, sram cell etc) and a network of interconnections. The circuit configuration required by the end-user is described using *Hardware Description Languages (HDL)*, such as *VHDL* and *Verilog*. In the case of Xilinx® FPGAs, using another program (*ISE Design Suite*) the code is converted into a bit-stream which is sent to the FPGA in order xto program it.

The method used for physically programming the FPGA depends on how the switches are implemented. Nowadays the most common methods are:

- **Antifuse** The switches are antifuses, devices which are in open circuit state until a high voltage is applied. Then they become short circuits, so a connection is established. These FPGAs can be programmed only one time.

- **SRAM** The switches are pass transistor or multiplexer. The state of each switch is stored in a SRAM cell. Xilinx® FPGAs are SRAM FPGAs.

- **FLASH** The switches are floating gate transistors which can be turned on or off injecting charge.

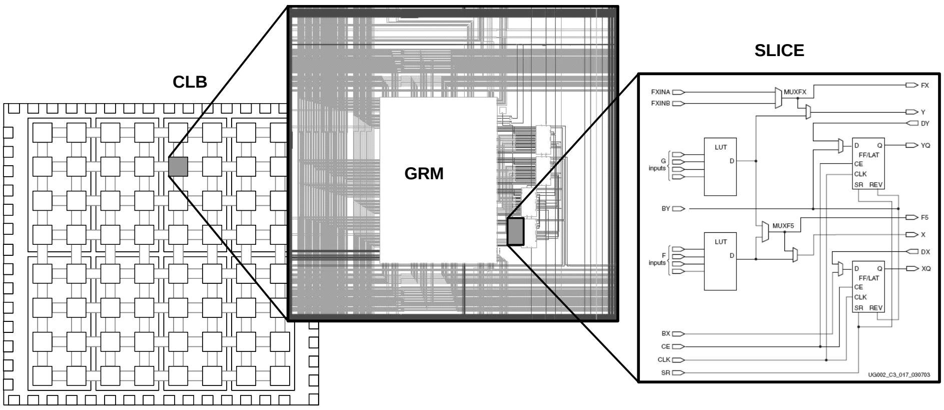


Figure 1.1: Basic structure of an SRAM-FPGA (*Virtex*™ *Family*)

A design based on FPGA doesn't allow to obtain an optimized result in power consumption, speed performance or area occupation. Looking to these parameters an approach based on ASIC could be better, since the user can intervene at every project layer, from the architectural to the transistor one. However there is a problem related to the *cost* of the device. Making a single device using an ASIC or a full custom approach can lead to an extremely high cost, while using a single FPGA the major part of the expense is that for the FPGA device. The development cost indeed is very expensive (and long) with full custom approach, while with the FPGA approach the developer has almost only to write the HDL code and test it. Nevertheless if the total number of devices is very large, an ASIC approach is better. In Figure 1.2a, the differences between ASIC, full custom and FPGA approaches are reported. In Figure 1.2b we show the cost trends for a single device made using designs based on ASIC and FPGA. What we can see is the fact that with low production volumes (less than 5000 pieces) a product based on FPGA is cheaper than other approaches. If the volume is bigger, the cost per single device drops a lot using an ASIC approach, since the NRE[3] costs are now spread on all the devices.

1.4 Why Temperature Matters?

There are many fields where electronics is employed and these fields have different characteristics. The radioactive environment is different in avionics, aerospace and automotive. In the orbits near the Earth for example there are more electrons and protons than neutrons, since they are trapped in Van Allen belts. In the atmosphere (as better described in Section 2.1) neutrons are the majority radiation type, so they are more relevant for avionic and automotive applications.

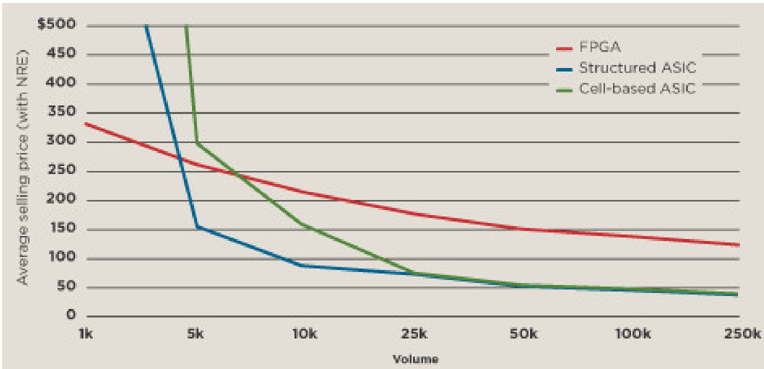
Another important fact is related to the conditions in which electronic components act. Humidity, dust and mechanic vibrations can lead to problems or malfunctions. One of the most critical factors is temperature: it affects the instantaneous functionality and also the characteristics degradation. The work is focused on the instantaneous dependence between SEU sensitivity and temperature in SRAM (in Section 3.4 we'll show the present knowledge about the temperature effects on SEU sensitivity).

Another problem related to temperature is that it is not always at the same value during the device lifetime. For example an aeroplane moves from the ground to high level in the atmosphere, with a temperature range from $-50\,^{\circ}$C in flight to 20-30 °C on the ground (see Figure 1.3). In the case

[3]NRE stands for *Non-Recurring Engineering*. This concerns the cost for the research, development and test of a new product.

SoC Design Options			
Feature	Full Custom	ASIC	FPGA
Masks Customized	All	A Few	None
NRE	$1M and up	$100K-$1M	$10K-$300K
Development Time	2+ years	About 1 year	Few Months
Unit Cost (10M/yr)	Small	Medium	Large
Unit Cost (100K/yr)	N/A (volume too low)	Medium	Larger
Unit Cost (1k/yr)	N/A	Volume too low	Largest

(a) Characteristics comparison of approaches to SoC design



(b) Trend of the cost of single device varying the production volume

Figure 1.2

of automotive, the problem seems not to exist. However in a car there are other heat sources. If an electronic device is located near the engine, the temperature in that place can achieve more than 100 °C when the car is running, or drop below 0 °C if it is parked outside. In all these scenarios the device has to work, so it's important to understand the effects of the temperature on SEU sensitivity in order to avoid malfunctions that can be also critical.

1.5 Motivation (Power Dissipation)

Considering only digital electronics devices, if the circuit implemented uses high frequencies or occupies a large area or the cooling system doesn't work properly, the internal power dissipation can lead to very high die temperatures.

In an electronic circuit the dissipated power is composed of two part. The first (and the larger) one is related to the *dynamic behaviour* of the circuit, that is the charging and discharging of the capacitances in the circuit. The

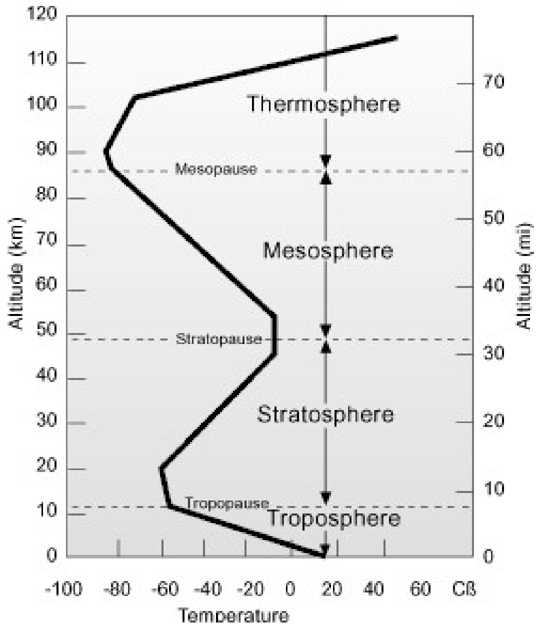


Figure 1.3: Atmosphere temperature for different heights

equation for describing this phenomenon is

$$P_{\text{diss}} = \alpha f_{\text{CLK}} C_L {V_{DD}}^2 \tag{1.1}$$

where f_{CLK} is the working frequency, V_{DD} is the supply voltage, C_L is the load capacity and α is the so-called *switching activity*. The latter expresses the fact that the circuit capacitances are not charged and discharged at every clock cycle, but they can remain in the same state for more than one clock period, for example because the input doesn't change or because it changes without influencing the output value. In view of that if the capacity loads are for example charged/discharged *on average* every 4 clock cycles, α will be equal to 1/4.

The other component of dissipated power is the *leakage power*. Many phenomena contribute to that: *Drain-Induced Barrier Lowering (DIBL), tunneling through the gate oxide, body effect, Gate-Induced Barrier Lowering (GIBL), etc.* A better explanation of these effects and the techniques used for lowering this power component are exposed in [3].

1.6 Research Goals

The basic idea behind this work is to study which effects the circuit frequency can induce on SEU rate in an SRAM. As shown in Section 1.5

a frequency increase causes an increase of power dissipation. The dissipated power can be measured monitoring the temperature, so we expect a temperature rise with a dissipation increase. The experimental relation between temperature and frequency is going to be shown in Section 5.1.

The effects of this induced temperature rise on SEU rate are not clear yet: as we will show in Section 3.2, there are many mechanisms involved in SRAM working. So theoretically predicting the right SRAM behaviour modification is difficult. Moreover the results are not easily explicable, as we are going to show in Section 5.2.

In the next chapters we shall explain better the radiation (Chapter 2) and the features of the electronic device (Chapter 3) involved in this experiment. We are going to show also the experiment setup (Chapter 4) built for achieving what we have described in this section. The results are shown in Chapter 5. We'll conclude with some considerations in Chapter 6.

2. Neutron-induced Radiation Effects

Information in this chapter are mainly taken from [4].

The neutron radiation results from the interaction between high energy particles ($E \gg 1\,\text{GeV}$), coming from the outer space, and the Earth atmosphere components, such as oxygen and nitrogen. These cosmic particles, also called *cosmic rays*, were released from the Big Bang or from Super Novas explosions. Another source of these energetic particles is the sun, but these particles have less energy ($E < 1\,\text{GeV}$).

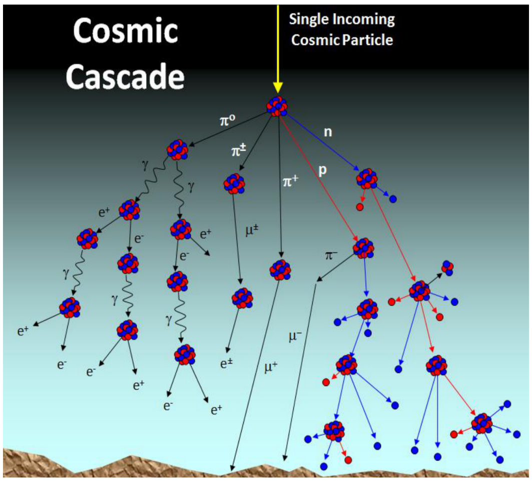


Figure 2.1: Reactions cascade by cosmic rays

When they enter the atmosphere, complex reactions cascades are produced (see Figure 2.1). As a consequence a lot of different particles are generated, such as protons, pions, muons, electrons and neutrons. Concerning the cosmic radiation, neutrons are considered the dominant source of SEE: this is due to their abundance compared with the other particles (see Figure 2.2).

2.1 Atmospheric Neutrons

The neutron radiation in the terrestrial environment can be characterized in many ways, but important aspects are its distribution in space and its spectrum (the distribution in energy).

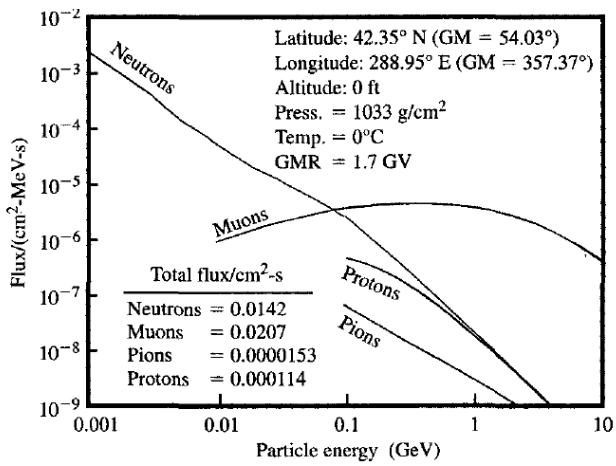


Figure 2.2: Terrestrial particle flux as a function of energy

2.1.1 Space Distribution

The magnetic field and the atmosphere of the Earth are the main factors which modulate the presence of the neutrons on our planet.

The magnetic field (Figure 2.3) is responsible for the shielding against the cosmic rays, which start off the cascades shown in Figure 2.1. This explains the dependence of the neutron flux on the *latitude*: as Figure 2.4a shows, the flux is higher at polar coordinates compared to the equatorial ones. This is due to the fact that the field lines are perpendicular to the incoming cosmic radiation at the equator, as depicted in Figure 2.3; instead at the poles they enter the Earth, so the shielding effect is reduced.

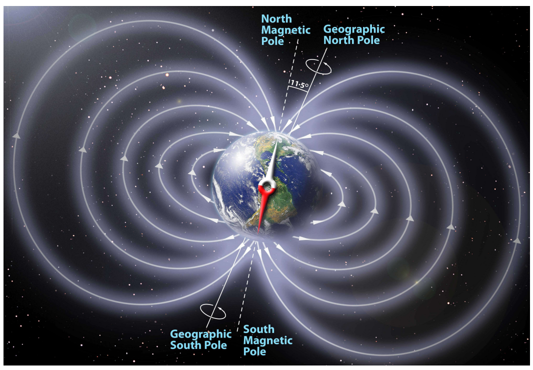
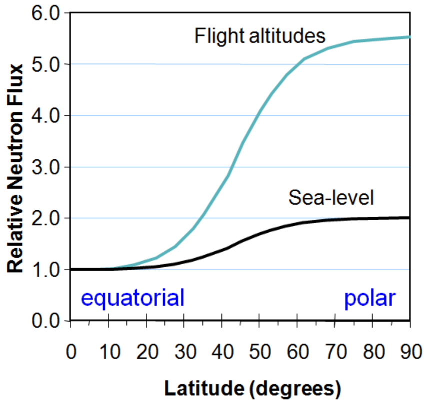


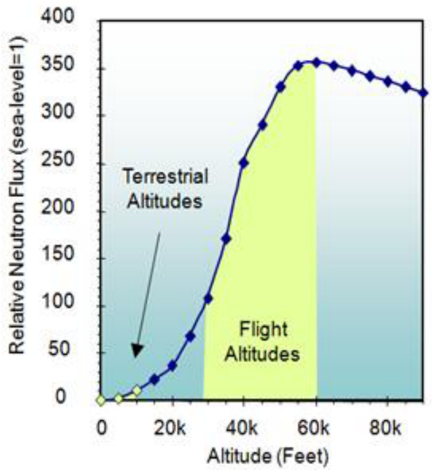
Figure 2.3: Lines of Earth's magnetic field

The neutron flux does not depend on *longitude*. The only exception is the South Atlantic Anomaly (SAA), located about over South Brazil. In this region the magnetic field is closer to the Earth, because of the not correspondence between the rotation axis and that of the magnetic field. The consequence is that the radiation can achieve a shorter distance from the Earth.

The dependence with the *altitude* is due to the interactions with the atmosphere components. Going from high altitudes to the sea level it is



(a) Neutron flux variation with latitude



(b) Neutron flux variation with altitude

Figure 2.4

obvious that the neutron flux drops: the neutrons indeed can be absorbed during the collisions with atmospheric atoms. However there is not only this phenomenon. As we have already seen, when cosmic rays hit the upper part of the atmosphere numerous particles are generated (Figure 2.1). Even though an intense proton flux is created, most of this flux is converted into neutrons and other particles while the protons are falling: this is due to nuclear reactions with oxygen and nitrogen and it's eased because of the charge of protons. These complex processes give as result a trend, shown in Figure 2.4b, where there is a peak around 18 000 m (around 60 000 feet), called *Pfotzer maximum*: here the neutron flux trend is hundreds of times larger than that at sea level. As shown in the figure, the region where aeroplanes flight includes this peak and it is characterized with high levels of neutron radiation, so studying the interactions of electronics with neutron radiation is important.

2.1.2 Neutron Flux Spectrum

The energies achieved by the neutrons at sea level are modulated by the interactions with the atmosphere constituents. As shown in Figure 2.5 there are three peaks in the neutron spectrum. Each of them is related to a particular nuclear reaction.

- The first peak at low energy stands for the neutrons which thermalized by elastic collisions with oxygen and nitrogen nuclei;
- In the middle we find the neutrons which collided in an inelastic way

and with resonance reactions with oxygen and nitrogen nuclei;

- In high energies range there are the neutrons which are produced in direct or knock-out reactions: the nucleons are emitted with approximately the same kinetic energy of the hitting neutron.

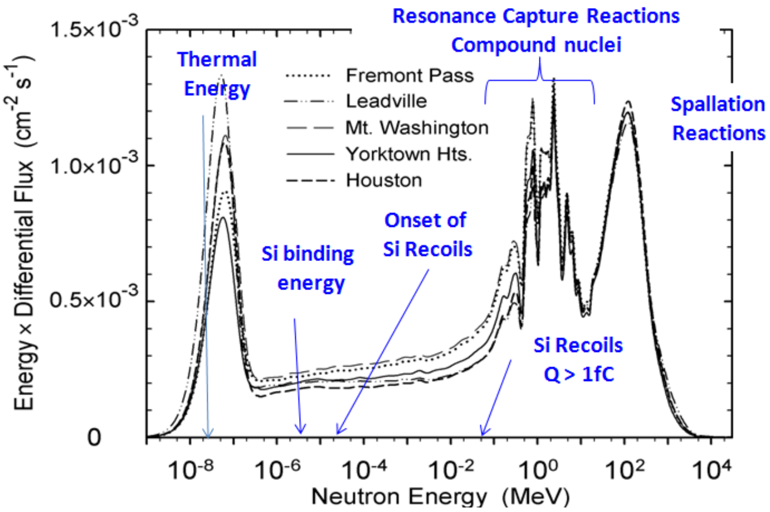


Figure 2.5: Neutron spectra for various geographical sites

In the next sections we shall explain how the interactions neutron-matter change varying the neutron energy.

2.1.3 Solar Activity Influence

The solar activity is another factor which has to be taken into account for modelling the neutron flux. The activity has a 11-years cycle during which the sun passes through very active and quiescent periods. When the activity is high, the amount of solar flares and corona mass ejections increases. This means the emission of x-rays and gamma radiation and plasma clouds at high temperatures in the space. These clouds are made up of electrons, ions and atoms. The electromagnetic radiation reaches the Earth before the plasma and it increases the atmosphere shielding effect, ionizing the upper part of the atmosphere. Therefore an electrostatic repulsion is established and the particles of the clouds are repelled. In such a way the proton flux is reduced and, as we showed in Section 2.1.2, the consequence is a reduction of the neutron flux. However if the solar flares or the coronal emissions are very rapid, the ionosphere has not enough time for being fully ionized. Therefore the repelling effect explained above is not activated.

2.2 Neutrons Interaction with Matter

Neutrons can't ionize directly the material which they hit, since they lack any kind of charge. However they can generate products that can ionize the material. These products depend on the interactions between neutrons and the matter and these interactions depends on the energy of the neutrons. In view of the different energies assumed by the neutrons (see Section 2.1.2), we need to understand which kinds of interactions there are and at which energies they can take place.

2.2.1 High Energy Neutrons

As we can imagine, if a neutron has a great amount of energy, it could move an atom located in a lattice. Since the main material used for semiconductor devices is *silicon*, it is important to study what could happen between a neutron and a silicon atom.

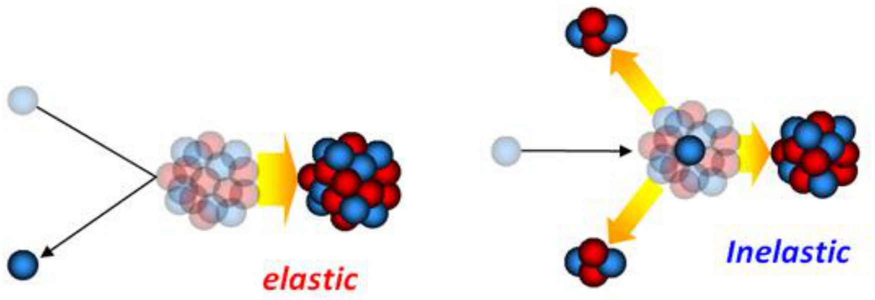
The first type of interaction is the *elastic scattering*, depicted in Figure 2.6a. Without considering relativistic effects[1] we can visualize the neutron and the silicon atom as two pool balls. In this way simple kinematic equations about linear momentum and kinetic energy can be solved. What is found is that

$$E_{kSi} \simeq 0.13 E_{kn}$$

where E_{kSi} and E_{kn} are the kinetic energies, after the collision, of the silicon atom and that of the neutron respectively. Since the energy required for the creation of a silicon vacancy is around 4 eV, a neutron has to have at least a kinetic energy of 30 eV for generating a vacancy in silicon lattice. If the neutron has an energy greater than 30 eV, the surplus acquired by the silicon atom is lost in electronic processes, such as the creation of electron-hole pairs. For example a surplus of 60 keV creates around 1 fC of charge that is the Q_{crit} for the SRAM in 45 nm technology.

The other interaction is the *inelastic scattering* (see Figure 2.6b). This happens when the neutron energy is great enough for exciting the nucleus of silicon atoms. As a consequence the wavelength associated to the neutron has to be comparable to nuclear dimensions and this occurs starting from 2 MeV (at 16 MeV it is as long as the silicon nucleus diameter). In this way interactions with multiple nucleons are established and the result is the generation of a new nucleus which is in a high energy state. When this nucleus decays, there is a release of nucleons and other nuclear fragments. Increasing the energy of the incident neutron, the number of reactions increases.

[1] This can be done considering neutron energies up to 2 MeV.



(a) Elastic scattering

(b) Inelastic scattering

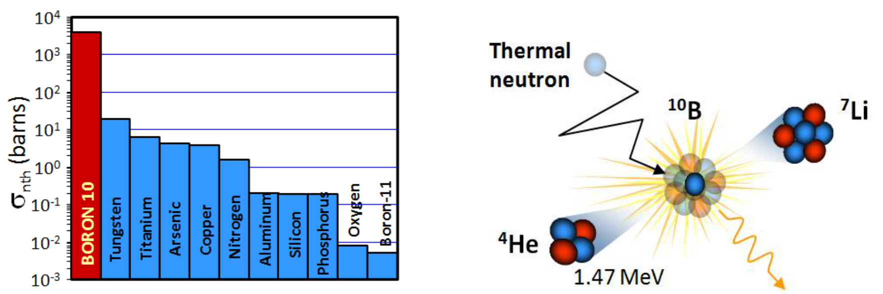
Figure 2.6

Looking at Figure 2.5 and at what we wrote about that, the neutrons of the high energy peak and of that in the middle are energetic enough for generating what we have described in this Section.

2.2.2 Low Energy (Thermal) Neutrons

If the neutrons have not enough energy for interacting with silicon atoms, it doesn't mean they can't cause problems to electronic ICs. They can't induce the generation of secondary products from silicon atoms, but they can interact with other elements in the IC die. Their energy is so low that they can be "absorbed" in a nucleus: as a consequence the latter becomes unstable and then it can emit electromagnetic radiation or decay, cracking into other particles.

One of these problematic atoms is an isotope of *boron*: ^{10}B. It is characterized by a great probability (also called *cross section*, see Section 2.4) to attract thermal neutrons as shown in Figure 2.7a. This probability is 3 to 7 orders of magnitude higher than that of other elements.



(a) Thermal neutron cross section of some nuclei

(b) Nuclear reactions caused by thermal neutron in boron nucleus

Figure 2.7

When a neutron is captured by a ^{10}B atom, the latter becomes unstable and it cracks, producing an alpha particle and a lithium ion, as illustrated in Figure 2.7b. The reaction products are ionizing particles, so they can generate amounts of free charge in the die: the alpha particle can generate 15 fC/µm and for the lithium ion that value is around 22 fC/µm. The range for these particles is 5 µm for the alpha particle and around 2 µm for the lithium recoil. If this charge is collected in some nodes, an SEE can be induced.

In ICs the boron is used as a p-type dopant and implant species in silicon and in the dielectric layers because of the use of borophosphosilicate glass (BPSG). Between these source of ^{10}B, the richest is the BPSG, since in the diffusions and implants the major part of the boron is the ^{11}B isotope. So the removal of the BPSG is a good way for mitigating the SEU sensitivity caused by ^{10}B presence.

2.3 Neutrons Effects on Electronics Devices

In Section 2.2 we showed the possible interactions between neutrons and matter. We can distinguish two kinds of induced problems. The first one regards the vacancy-interstitial generation. We saw how high energy neutrons interact with silicon atoms: when they hit a nucleus, this is moved from its place in the lattice, creating a vacancy (see Section 2.2.1). The interstitial is referred to the moved atom which is no more part of the lattice. This damage reduces the carrier mobility in the semiconductor or creates intermediate levels in the gap energy band. A great amount of this kind of damage degrades the device characteristics, leading to a possible device failure. A way for partial repairing these damages is the *thermal annealing*: the device is heated and the interstitial can move and recombine with the vacancy or other defects [5].

The second kind of problem is related to the generated charge in the device: we have seen how a neutron can indirectly create charge in Section 2.2. The effects on electronics are better explained in the next section.

2.3.1 Single Event Upset - SEU

An SEU is a soft error where a bit value stored in a memory cell (e.g. SRAM, DRAM etc) changes because an energetic particle generates an amount of charge great enough to provoke the reversing of the circuit transistors state.

The generated charge is not totally involved in the bit-flipping: first of all it's shared among different device nodes and then recombination removes a part of it. A such approach leads to conservative constraints. For these reasons the *collected charge* Q_{coll} is taken into account. It depends on many factors, such as the size of the depletion region of the node, the biasing and the substrate structure.

To understand how much a device is sensitive to the collected charge, we can define a threshold for Q_{coll} beyond that the device suffers a soft error. This threshold is called *critical charge* Q_{crit}. Q_{crit} has not a constant value, since it varies with the circuit sensitivity. Q_{crit} is a general concept so it can be used with other kinds of soft errors.

As we have noted in Section 2.2, neutrons can't generate charge directly, but they can interact with matter creating sub-products which ionize the device. Then the produced charge can be collected at some nodes and cause SEUs.

2.4 Cross Section

The *cross section* σ of a sample (device, circuit, board etc) is defined as the part of the device total area which is sensitive to the radiation we are considering. Mathematically if A is the sample total area exposed to the radiation, N is the total number of particles hitting the surface and N_{counts} is the number of damages caused by the radiation, then

$$\sigma = \frac{N_{\text{counts}}}{\frac{N}{A}} = \frac{N_{\text{counts}}}{\Phi} \quad \left[\frac{\text{counts cm}^2}{\text{particles}}\right]$$

where $\Phi = N/A$ is the particle *fluence*[2]. For the sake of simplicity we have considered that particles impinge perpendicularly.

The application of this concept concerns the fact that if we know the value of σ and the number of particles in the environment, we can evaluate a forecast about the number of damages or malfunctions that will occur. From a statistical point of view

$$N_{\text{counts}} = N\left(\frac{\sigma}{A}\right) = \frac{N}{A} \cdot \sigma = \sigma \cdot \Phi$$

since σ/A represents the probability for a hitting particle to generate a damage.

In our work the evaluated cross section is that about SEUs in the configuration memory (σ_{SEU}), so N_{counts} coincides with the total number of SEUs in the configuration memory (N_{SEU}).

[2]The fluence represents the number of particles which pass through an unit area. It is the integral of the *flux* with respect to the time, since the flux is the number of particles passing through an unit area per unit time.

3. SRAM Cell

3.1 SRAM Cell Structure

The basic SRAM cell is made up of 6 transistors (Figure 3.1). M_5 and M_6 are used to access the stored value through *Word Line* signal (WL), both for read and write operations. The other 4 transistors form two CMOS inverters, where the output of the first is the input of the second and vice versa: a regenerative feedback is established. Since this situation provides both the stored signal and its inverse, noise margins during the read and write operations are improved using those data as input of a sense amplifier.

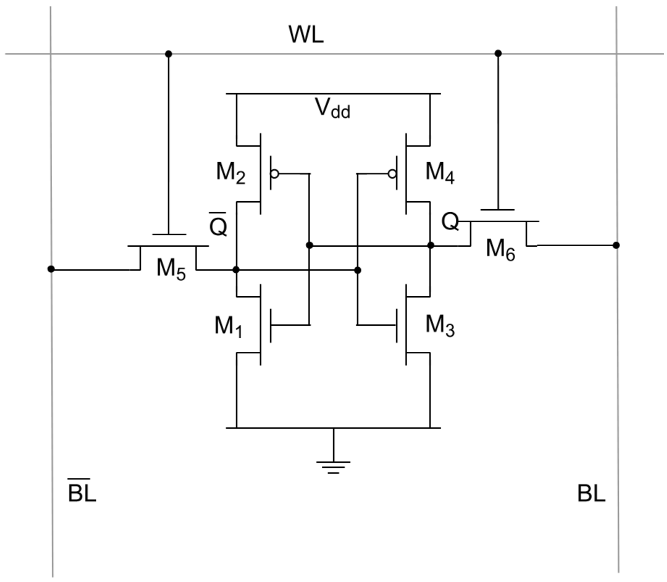


Figure 3.1: 6 Transistors SRAM cell structure

If a "1" is stored ($Q = 1$) the transistors M_1 and M_4 are turned on and the others are turned off. In the opposite case ($Q = 0$) M_2 and M_3 are off while the others are on.

The transistors are sized keeping in mind the circuit integration (transistors smaller) and the problems given by the read and write operations (transistors bigger). In the last case indeed when M_5 and M_6 are turned on for read operation, the bit line capacity load is connected to internal nodes, in which the bit is stored. If the SRAM transistors are too small, the stored value can flip due to the bit line precharge. Bigger transistors provide faster response to voltage variation of internal nodes.

Another way for building an SRAM cell is shown in Figure 3.2 and the aim is obviously to lower the area occupation. It is achieved decreasing the number of transistors used in the implementation. The negative consequence is the raise of the standby current.

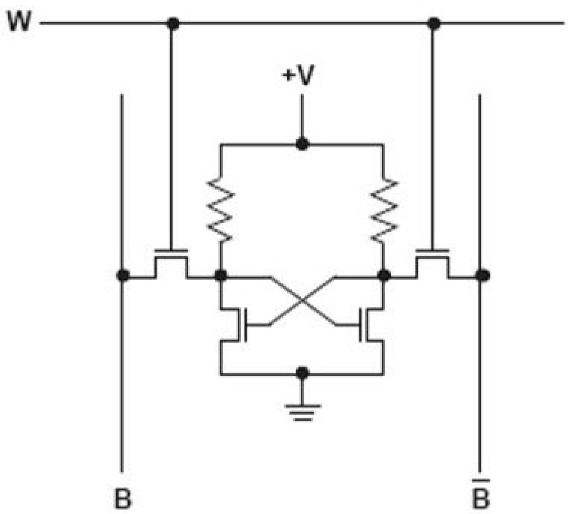


Figure 3.2: 4 Transistors SRAM cell structure

3.2 Radiation Effects on SRAMs

To illustrate which can be the effects of an ionization in an SRAM cell, we start the analysis supposing that the bit value stored is "1". As we have explained in Section 3.1, if $Q = 1$ then M_2 and M_3 are off and M_1 and M_4 are on.

If an ionizing particle hits the cell near the drain of M_3, an amount of charge is collected at this node: this is due to the electric field in that region. The collected charge is negative, since the drain of an N-MOS is N-doped and the pn junction between the drain and the body is reverse biased: the drain indeed is connected to the node Q.

This excess of charge drops the node voltage: on the one hand current passes through the P-MOS M_4 for restoring the previous voltage; on the other hand the feedback between the two inverters reduces the input voltage of the second inverter. Thanks to this last drop, the P-MOS M_2

starts to turn on, while the N-MOS M_1 is turning off. In this way the output voltage of the second inverter starts to raise, since a current is flowing through the P-MOS M_2. This increase contributes to change the states of mosfets in the first inverter and so on. As a result, if the initial voltage drop is enough *large* and *long*, a bit flip occurs.

With the last statement we have underlined the importance of two aspects, the *amplitude* and the *duration* of the voltage variation caused by the ionization. These two points can be analysed using the Q_{crit} concept, illustrated in Section 2.3.1. In the case of SRAM it's defined as

$$Q_{crit} = C_{node} \cdot V_{data} + \tau_{switch} \cdot I_{restore} \tag{3.1}$$

where C_{node} is the capacity load of one SRAM output node, V_{data} the data voltage margin, which is the difference between the switching threshold voltage and the data stored voltage, $I_{restore}$ the capability of the transistors to keep the node voltage at the correct value and τ_{switch} expresses the response speed to a circuit perturbation.

3.3 Hardening Techniques for SRAM

Information in this chapter are mainly taken from [6].

In Section 1.2 we have described which problems can affect electronic devices used in radioactive environments. When electronic components are employed in such environments, we have to avoid situations where errors occur. If these situations are impossible to prevent, we need to manage these situations in a proper way. In electronics many techniques have been developed for avoiding or for managing erratic situations. In this section we are going to show techniques used for avoiding error occurrences in SRAM cells.

Hardening the SRAM design includes all the approaches which can lead to a lowering of the memory sensitivity to SEU. In the last section we have listed the factors which influence Q_{crit} for SRAM: if we increase them, we can achieve a smaller SEU sensitivity.

About the first part of Equation 3.1, the simplest way to increase Q_{crit} is increasing the transistors size or the supply voltage: obviously these are not good operations for improving the integration and the power consumption.

The second part of Equation 3.1 instead is related to the dynamic behaviour and the two factors require different adjustments. $I_{restore}$ coincides with the current driven by the transistors and in the previous explanation of SEU in SRAM it has been the P-MOS M_4 drain current. The other factor τ_{switch} is increased if the time while the restoring current can act increases.

Lowering the SRAM speed is the way to increase τ_{switch} and it can be done reducing the feedback response. This concept is used in the *resistive hardening* (illustrated in Section 3.3.1).

An important consideration regards the scaling in the SRAM technology. As the technology improves, transistors size decreases and so the capacitance and the cell area. The supply voltage is reduced as well, lowering the power consumption. The reduction of cell volume helps to loose collection efficiency, but what happens to the capacity load and the supply voltage lowers the Q_{crit}. The result of the relation between these factors is not constant: indeed in the first generations the SEU sensitivity increased and then it dropped, as shown in Figure 3.3[1]. The drop is due to the saturation of supply voltage (around 1 V), decreasing in junction collection efficiency and increasing charge sharing phenomenon among neighbouring nodes.

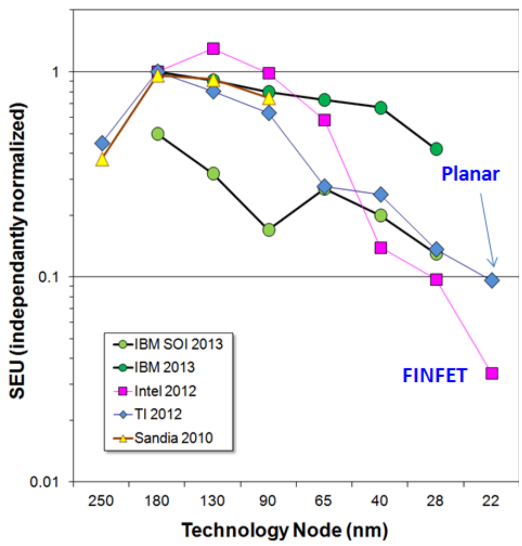


Figure 3.3: Normalized SEU for SRAM built in different technologies

3.3.1 Design Level

Resistor Memory Cell

This approach consists in acting on the Q_{crit} increasing the τ_{switch}, slowing down the feedback speed. In Figure 3.4 we can see the design which matches the illustrated purpose. The resistors inserted between the outputs and the inputs of each inverter increase the necessary time for the variation caused by the charge collection to propagate in the feedback.

Memory Cell with Different Feedback

In these designs (pictured in Figure 3.5) an appropriate feedback helps to restore the data if an SEU occurred. For example in the DICE SRAM (Figure 3.5b), the cell is duplicated (there are 4 inverters). The advantage is twin: on the one hand the value is stored twice (redundancy), so it provides

[1] The data in this figure doesn't take into account memories with BPSG as passivation layer.

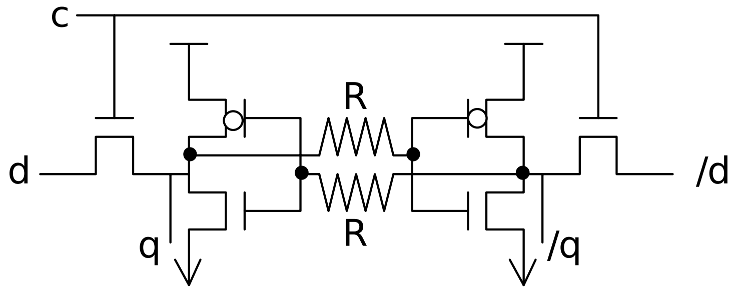
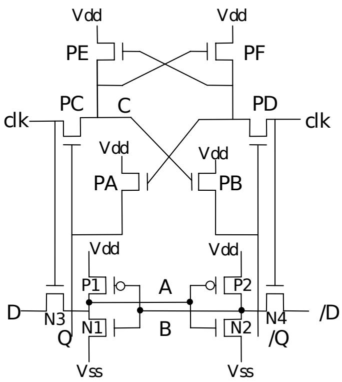
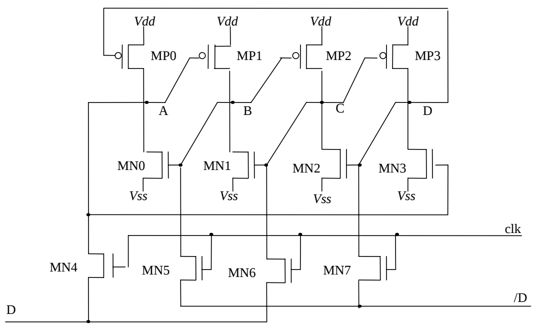


Figure 3.4: Resistive hardened SRAM cell

a source of uncorrupted data after a single event; on the other hand the uncorrupted section is used for recovering the corrupted data (feedback).



(a) IBM hardened SRAM cell



(b) DICE hardened SRAM cell

Figure 3.5

3.3.2 Logic Level

The previous rad-hardening techniques are useful only if we can operate at manufacturing layer. This is an expensive way for applying the hardening to our circuits. If we can't access this level, other techniques are required. In this context the *Error Correcting Code (ECC)* is a good way to lower the SEU sensitivity of our system. It is a technique where an algorithm based on encoding/decoding for the self-correction is used. The basic idea is to map the original data into longer data, increasing the redundancy and so improving the robustness. This process requires more memory for storing the "new" data. Examples of these algorithms are *Hamming code*, *BHC*, *Reed-Solomon code* etc. Simple codes can be implemented in hardware, using extra memory and circuit for encoding and decoding.

3.4 SRAM Cross Section Dependence on Temperature

The current knowledge about the dependence between SEU sensitivity and temperature in SRAM is exposed for example in [7]. The most remarkable result shown in [7] for our analysis is the irradiation of two memories with thermal neutrons at several temperatures. These two memories were made by different vendors but in the same technological node (180 nm).

What the authors observed was a different trend for the two devices. As reported in Figure 3.6 we note that in the range 25-85 °C the cross section for the *Device A* decreases (less than 7%), while in the case of *Device B* happens the contrary, increasing by around 15%.

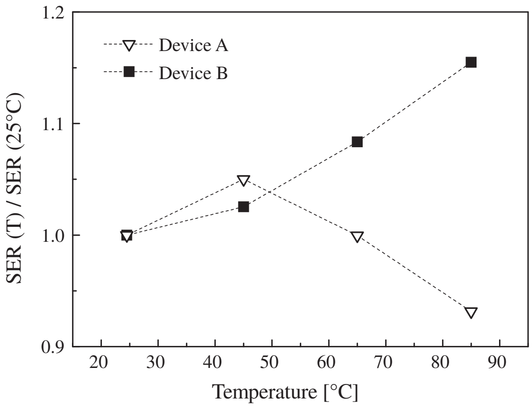


Figure 3.6: Experimental error rate for 2 SRAM chips after high energy neutrons exposure

The authors then have described which are the factors that influence the variation of SEU sensitivity. The analysis starts from Equation 3.1, considering each of its variables. Many elements were examined looking for a better description of the dependence between temperature and Q_{crit}. As shown in Table 3.1 there is not a defined and clear relation between SER and temperature, since some factors increase and others decrease with the temperature rising. These theoretical considerations and the experimental results showed above lead us to state that additional research is necessary.

Parameter	T Dependence	Impact on SER	Variation Range
Linear Energy Transfer LET	≈	≈	< 1.5%
Drain Current I_{DS}	↓↓	↑↑	< 20%
Cell Write Time t_{write}	↑↑	↓↓	< 8%
Peak Drift Current I_O	↓↓	↓↓	10-20%
Depletion Region Width x_d	↑	↑	< 2%
Funneling Length $I_{\text{funneling}}$	↑↓	≈	-3% to 3%
Diffusion Length L_{diff}	↑↓	↑↓	n.a.
Ambipolar Diffusivity D^*	↑↓	↑↓	-7% to 11%
Minority Carrier Lifetime τ	↑	↑	n.a.

Table 3.1: Factors influencing the SER with respect to temperature variation

4. Experimental Setup

4.1 Introduction

We have described the purpose of this experiment in Section 1.6. For achieving it we need an SRAM integrated in a digital circuit. We can find this situation in an SRAM-FPGA, as described in Section 1.3. The monitored SRAM is the configuration memory of the FPGA, which is placed in all the FPGA circuit.

The implemented circuit is a circuit whose working frequency can be controlled easily, without any input dependence.

We used a Zynq™ by Xilinx® as FPGA and the circuit was elaborated using Xilinx® software.

A part of the experiment was about the heating and the cooling from the outside. We performed that to check if the kind of heat source influences the SEU sensitivity.

4.2 Xilinx Zynq FPGA

In the experiment we used a low cost evaluation and development board called ZedBoard™ (Zynq™ Evaluation and Development) based on the Xilinx® Zynq™-7000 All Programmable SoC (see Fig. 4.1). Among the various board features the most important for our work was the USB-JTAG interface, used for writing and reading the configuration memory [8].

4.2.1 Zynq™-7000 All Programmable SoC

The Zynq™-7000 All Programmable SoC is a System-on-Chip in 28 nm technology with a dual-core ARM® Cortex™-A9 based processing system (PS) and a Xilinx programmable logic (PL).

In particular, the chip used in the ZedBoard™ is a Xilinx® Zynq™ XC7Z020-CLG484-1[1]. About the PL, we find a Xilinx® Artix®-7 with 85 000

[1]The first part of the code (XC7Z020) is referred to SoC components: concerning the PS only the operating frequency changes; looking to the PL, the Xilinx equivalent model

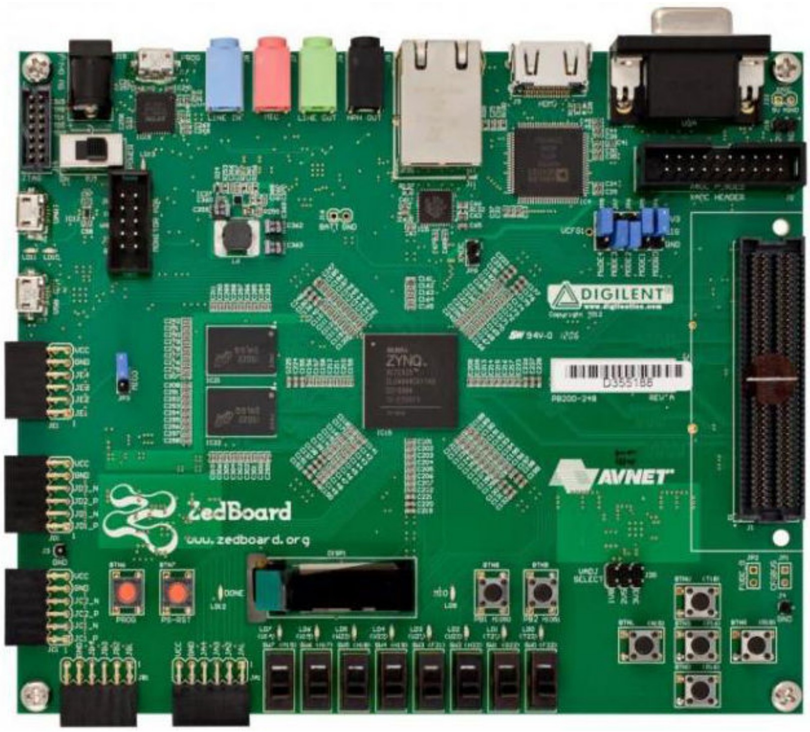


Figure 4.1: The ZedBoard™

programmable logic cells, 53 200 Look-Up Tables (LUTs) and 106 400 flip-flops [9]. The configuration bit-stream size is 32 364 512 bit [10].

It's important to understand that some features related to the specific programmable logic are not available in the Zynq™Soc, such as *SelectMAP*, a parallel configuration interface, while others are controlled through the PS (*cascade mode*) [10], for example ICAP (*Internet Content Adaptation Protocol*). The USB-JTAG interface allows to program and read the configuration memory of PL without employing PS, although its speed is lower than other interfaces.

An important feature we used is the temperature sensor. It's located in the middle of the PL, so it guarantees a good measure of the die temperature. The maximum measure error reported in [11] is ±4 °C: however this is the maximum error we can find in a single measure, it's not a constant error in each measure. So the sampled values aren't affected by this error.

differs among the different versions. The second part (CLG484-1) is referred to the device package [9].

4.3 Tested Circuits

The circuit we designed is a very simple digital circuit. We checked many possible designs, some of them showed in [12]. The purposes to satisfy were:

- Heating the die in a controlled way;
- Occupy as *much* area as possible;
- Using as *many* FPGA components as possible.

We decided to use *counters*, because they employ both LUTs (adders) and flip-flops (registers). The counters were designed as synchronous elements so changing the clock frequency, the power consumption (and the temperature) changes. About the counter length we chose 8 bit, because not all counter bits switch every clock cycle. While the *least significant bit* (*lsb*) value changes every clock cycle, the second bit switches half times less than the lsb. In this way the *most significant bit* (*MSB*) switches every 2^7 clock cycles. If we chose a greater length, the other bits would switch with very low frequencies, with a negligible contribute to the power dissipation. Another reason because we chose counters instead of other circuits is that this implementation allows to obtain a general working situation, since the switching activity is not the same for every gate or flip-flop in the circuit.

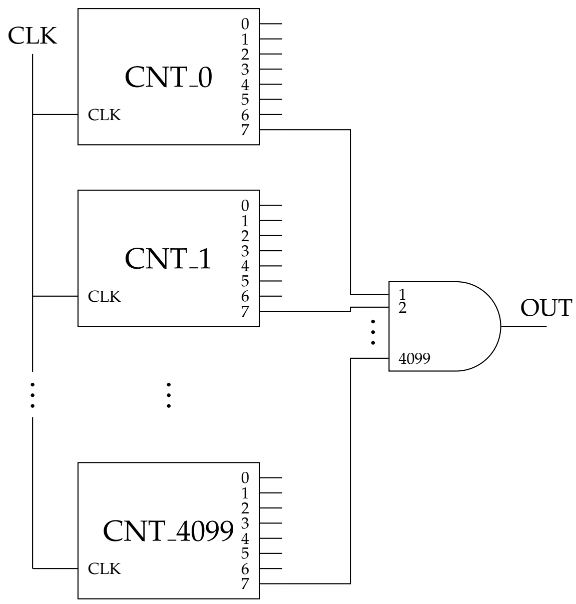


Figure 4.2: Schematic of implemented circuit

For obtaining a correct circuit an output is required, otherwise during the synthesis and the place-and-route the circuit is simplified and nothing is created. As a consequence we took the MSB of every counter and connected them to an AND gate. The AND output was sent to an external pin of the board. It was used only for the described purpose and for checking that the circuit worked with the designed frequency. The circuit we obtained is shown in Figure 4.2.

The circuit synthesis, place-and-route and creation of configuration files were performed using Xilinx® ISE Design Suite™. For easily managing the clock frequency an ISE utility was employed (*IP CORE Generator*). The *Clocking Wizard* was used for configuring an IP component. It sets up a *mixed-mode clock manager* (MMCM) and with that utility we could set the frequency for the circuit. The available range is from 10 kHz to 800 MHz.

As explained in 4.1 the goal is to check the sensitivity of the SRAM configuration memory. Since the configuration bit-stream affects the circuit layout, it was important to maintain the same circuit in all designs. Therefore using a particular functionality in ISE Design Suite, we could keep the same placed-and-routed circuit in every design, changing only the frequency[2].

Looking to the Equation 1.1 the power dissipation (and so the die temperature) depends on 3 factors: the working frequency, the switching activity, the load capacity and the supply voltage. So for achieving different temperatures we should change these factors. Considering that we had the same circuit among all the designs, the load capacity and the switching activity were fixed. Thereby we could change only the working frequency for varying the temperature, as the supply voltage is not alterable easily by an end-user.

In Table 4.1 we show the features of the implemented circuit. The frequency range is between 100 MHz and 700 MHz with a step of 100 MHz. In this way we obtained 7 designs.

Counters	4100
LUTs employed	33 571 out of 53 200 (63%) 29 471 used as logic 4100 used as route-thrus
Flip-Flop employed	32 800 out of 106 400 (30%)

Table 4.1: Data about the designed circuit

[2]Only the configuration bits of the clock manager IP change.

4.4 Test Setup

For the experiment we had 3 ZedBoard. We decided to use all of them at the same time and the structure shown in Fig. 4.3 was built. This allowed us to limit the statistical error. Then it was placed with other experiments on the beam line as shown in Fig. 4.4. In this way during the experiment 3 different designs run at the same time and they were managed by a simple script.

Figure 4.3: The ZedBoards in parallel

4.4.1 Script and Comparison Program

The script was a simple *batch* script. The sequence of pseudocode instructions is shown in Algorithm 1.

It's important to note two facts about the exposure time:

1. The exposure time starts with the FPGA programming and ends with the FPGA data reading: in the case of `FPGA1` the time regarding the data reading of `FPGA2` and `FPGA3` is not part of `FPGA1` exposure time. About `FPGA2` there is no exposure during the data reading of `FPGA3` and the programming of `FPGA1`. For `FPGA3` the programming times of `FPGA1` and `FPGA2` don't contribute to its exposure time.

2. The exposure time is different among the boards: for `FPGA1` the total exposure time is made up of the programming times of `FPGA2` and `FPGA3` and the Δt. For `FPGA2` it's composed of programming time of

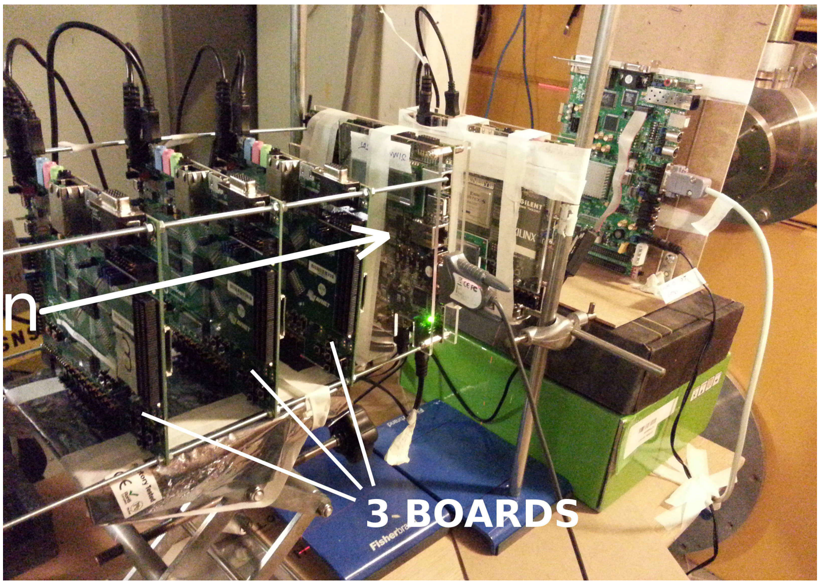


Figure 4.4: The ZedBoards with other experiments on the beam line

FPGA3, the Δt and the readback time of FPGA1. In the case of FPGA3 there are the Δt and the readback times of FPGA1 and FPGA2.

For every FPGA we considered also an addition of half program time and half data reading time for the total exposure time. This was done because the read and write operation are done bit per bit and during this time the memory is exposed to the radiation. Considering average values, we can imagine that half of the total SRAM cells are read or written in the first half part of the time and the rest in the second period. So the cells start to be exposed from beginning of the second half of the operation time. In Table 4.2 the various times are shown.

For checking errors, we compared the file created from the readback with a golden file. Each one is a *.bit* file, which contains the bit stream to configure the device. So these files are sequences of bits and they can be compared bit-a-bit. This task was carried out by a little C-program: it printed the errors[3] in a *txt file* with other information, such as the date, the hour and the temperature, sampled before and after the readback. This was done also for controlling the right execution.

[3]If a bit in the file from the readback is different from the corresponding bit in the golden file, the program printed the address of the erroneous bit.

Algorithm 1 Pseudocode of experiment script

```
procedure NEUTRON RADIATION (FPGA1, FPGA2, FPGA3)
    repeat
        Program FPGA1
        Program FPGA2
        Program FPGA3
        Wait Δt
        Read_Temperature FPGA1                ▷ FPGA1 data reading
        Readback FPGA1
        Read_Temperature FPGA1
        Compare_Bit_File_of FPGA1
        Read_Temperature FPGA2                ▷ FPGA2 data reading
        Readback FPGA2
        Read_Temperature FPGA2
        Compare_Bit_File_of FPGA2
        Read_Temperature FPGA3                ▷ FPGA3 data reading
        Readback FPGA3
        Read_Temperature FPGA3
        Compare_Bit_File_of FPGA3
    until Experiment ends
end procedure
```

4.4.2 Timing

The die temperature has not always the same value: during the data reading time the FPGA is not powered, so the temperature drops 4-5 °C (during the program operation the FPGA is powered). This problem is solvable if that time becomes negligible with respect to the total exposure time. As consequence the time interval Δt is necessary and it has to be set to guarantee that the total exposure time is at least around 10 times the temperature rising time.

We noted that, after a readback, the temperature rising time was around 30 seconds. But, as described above, the beginning of exposure time for an FPGA is not straight consecutive to the end of that FPGA readback. Therefore the time during which the FPGA is exposed to neutrons and the temperature is lower than that at steady state, is lesser than 30 seconds. We are going to show how much time is spent by the FPGA during the exposure and at a lower temperature than that at steady state:

- `FPGA1`: in this case the exposure starts after the readbacks of the other two FPGAs. So in the worst case the temperature rising finishes after 2-3 seconds from the exposure beginning.

- `FPGA2`: for this FPGA the exposure starts after the readback of `FPGA3` and the program of `FPGA1`. So the temperature rising finishes after 7-8 seconds from the exposure beginning.

- `FPGA3`: here the exposure starts after the program of the other two FPGAs. So this is the most critical case, because the temperature rising finishes after 14-15 seconds from the exposure beginning.

In Table 4.2 the characteristic times are reported. From the comparison between these values and the data written above, we can see that the timing constraints are maintained.

Instructions Set	**Duration (s)**
Program	8
Data Reading	15
Δt	150
Exposure `FPGA1` per cycle	177.5
Exposure `FPGA2` per cycle	184.5
Exposure `FPGA3` per cycle	191.5
Exposure `Single FPGA` per cycle	161.5

Table 4.2: Characteristic times in the experiment

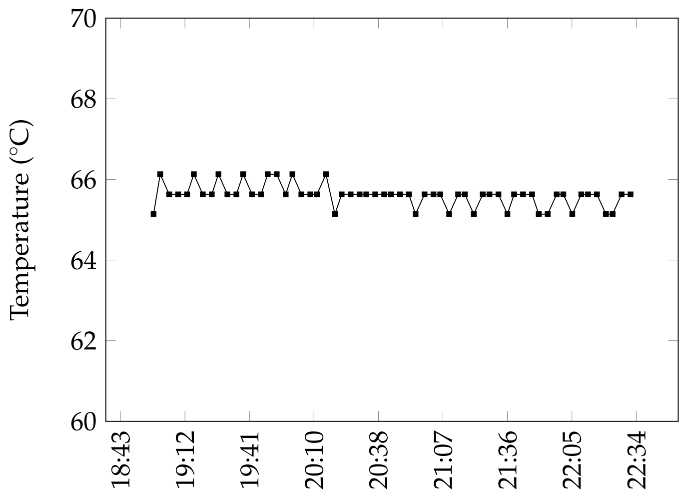


Figure 4.5: Example of temperature trend during the experiment

4.4.3 Cooler and Heater Setup

With the basic setup showed in Section 4.4 another experimental setup was created for heating and cooling the FPGA from outside.

For heating we placed a resistor in front of the FPGA: we could modulate the temperature changing the applied voltage. For cooling the FPGA we employed a *Peltier cell*. It's a semiconductor device (see Fig. 4.6) where the heat is transferred from a surface to the other, depending on the applied voltage (*thermoelectric effect*) [13]. The cell is usually packaged with ceramic material. The heat generated by the FPGA is removed and it's dissipated on the other surface using a CPU heat sink with a fan, as showed in Fig 4.7.

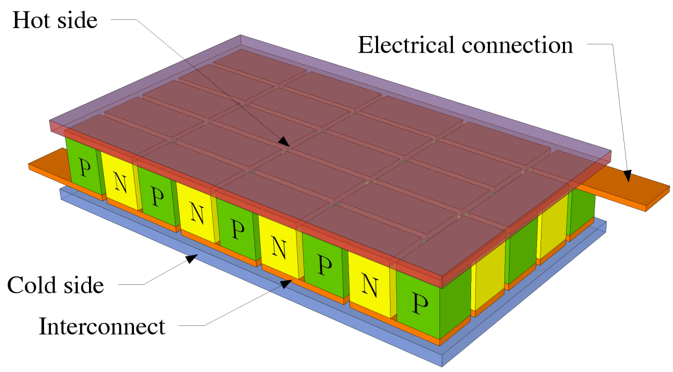


Figure 4.6: Draw of a Peltier cell

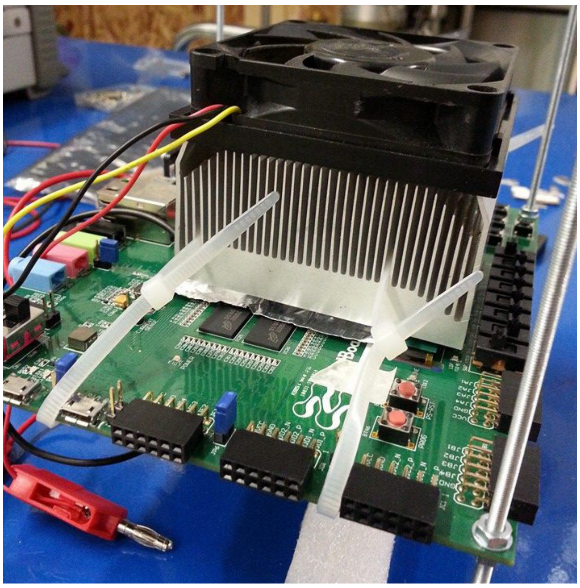


Figure 4.7: Setup with the Peltier cell and the heat sink

4.5 ISIS Facility

The experiment took place at the *Rutherford Appleton Laboratory* on the *Harwell Science and Innovation Campus* in Didcot, United Kingdom. The ISIS facility provides a neutron beam with good characteristics to study materials at the atomic level for several kinds of research, from physics to engineering to geology etc.

4.5.1 Neutron Spectrum

The process for producing the neutrons for the irradiation is based on the so called *spallation process*, where a heavy-metal piece is bombarded with energetic protons. From these collisions, neutrons are released from the nuclei of the target heavy-metal atoms.

In ISIS, *tungsten* is used as metal target and an *aluminium oxide foil* as proton source: H^- ions are accelerated in a linear accelerator and sent against the foil. The stripped protons are then injected in a synchrotron and they arrive at the target as pulses. The produced neutrons are then reduced in energy using a *moderator*. There are many lines which exploit this neutron beam [14]. In our experiment the line called VESUVIO (in *Target Station 1* Fig. 4.8) was used: it provides neutrons above 1 eV (*epithermal neutrons*).

Figure 4.8: ISIS experimental hall for target station 1

An important aspect of these plants is the energy spectrum of the beam. In Fig. 4.9 some neutron spectra are reported: in this chart we can see and compare the ISIS spectrum with other facilities spectra, but the most relevant is the comparison with the sea level neutron flux. The latter is not the energy of the real flux, but it's multiplied by 10^7 or 10^8. This comparison is important because in this way accelerated tests can be conducted: more similar the spectra, more accurate the accelerated tests. The available neutron flux was of about 5×10^4 neutrons/(cm^2 s) for energies above 10 MeV. The beam was focused on a spot with a diameter of 2 cm plus 1 cm of penumbra. Irradiation was performed with normal incidence.

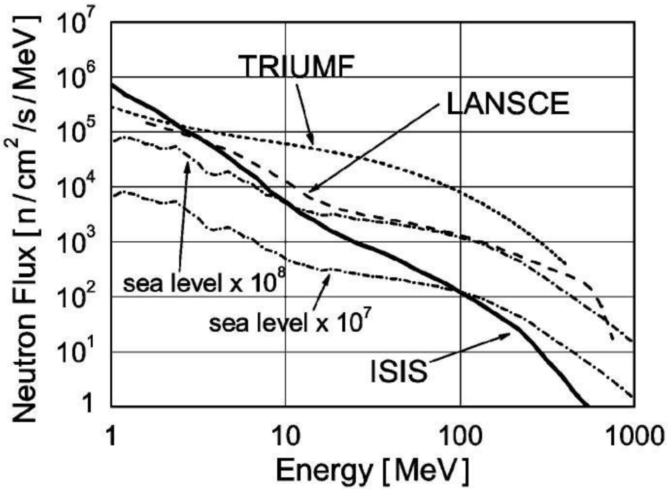


Figure 4.9: Comparison between spectra of some neutron facilities

5. Experimental Results

5.1 Power and Frequency

As reported in Section 1.5 the dynamic power dissipation in a CMOS circuit is expressed by Equation 1.1

$$P_{\text{diss}} = \alpha f_{\text{CLK}} C_L {V_{DD}}^2$$

We have built every design in such a way to keep constant the load capacity C_L, the supply voltage V_{DD} and the switching activity α (see Section 4.3). The relation between P_{diss} and f_{CLK} is linear. There is then a *leakage* component of the dissipated power, but we can assume it is constant if the supply voltage and the temperature are constant. The power is dissipated through the device, increasing its temperature. Therefore we can check the temperature to obtain an estimate of total power dissipation.

Design (MHz)	**Temperature (°C)**
700	65
600	62
500	58
400	53
300	49
200	44
100	40
700 H	81
100 H	66
600 C	42
100 C	24

Table 5.1: Designs temperatures without and with heater (H) or cooler (C)

In Table 5.1 we have reported the data about the temperatures achieved by our 7 designs without any cooler or heater. In addition the temperatures achieved using heater (H) or cooler (C) are reported in the same table. In

Figure 5.1 we have plotted the data included in the first part of Table 5.1 (without heater or cooler) to show the linear relation between the dissipated power and frequency: we can see how the points are almost aligned along a straight line. This line can't pass through the axis origin because of the contribution of leakage component.

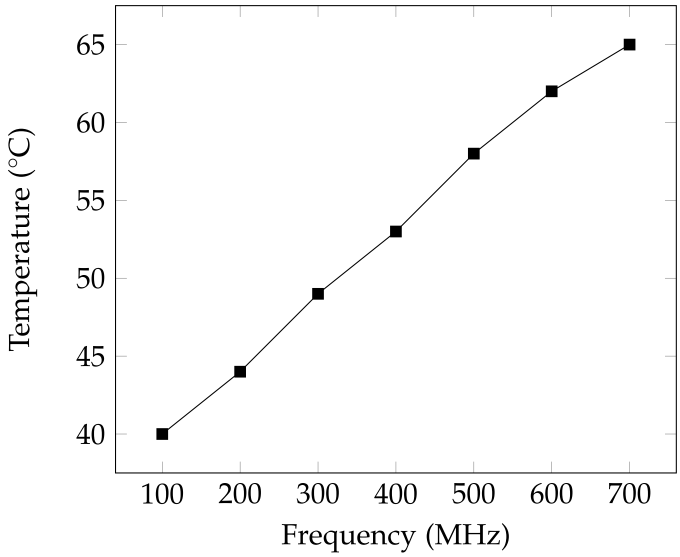


Figure 5.1: Power dissipation (temperature) as function of frequency in our experiment

5.2 Experimental Atmospheric Neutrons Cross Section

The data exposed in Table 5.2 were elaborated using information about the number of errors and the flux. The first information was extracted from the *txt files* illustrated in Section 4.4.1. The other information were included in files provided by the ISIS team. The fluence (used for obtaining the cross section as expressed in Section 2.4) was calculated using the data about the average flux on an hour: in this way it represents the integral of the flux with respect one hour. The average flux was calculated from samples taken every 3-4 s.

The errors reported were polished up looking at many things:

- **Beam Status:** if the beam was turned off or was too weak, the errors in that period of time were discarded;
- **Proximity:** if the errors addresses in the memory were found too close[1], only one of these errors was counted. This reasoning is con-

[1]The distances in the addresses are often the same for these *wrong* errors.

nected to the fact that, having a long exposure time (see Table 4.2) and the high flux available at ISIS, it may occur that more than one neutron corrupt the FPGA cross section in one single readback. However when many errors occur close to each other and at a repeated distance, it indicates that they are not different events, but they are generated by the same neutron. When errors observed in the same readback are found to be close to each other, they are considered a single event.

Design (MHz)	Cross Section per Bit (cm^2/bit)
700	1.77×10^{-15} ($\pm 0.89 \times 10^{-16}$)
600	2.31×10^{-15} ($\pm 1.2 \times 10^{-16}$)
500	2.17×10^{-15} ($\pm 1.1 \times 10^{-16}$)
400	2.18×10^{-15} ($\pm 1.1 \times 10^{-16}$)
300	2.00×10^{-15} ($\pm 1.0 \times 10^{-16}$)
200	1.99×10^{-15} ($\pm 1.0 \times 10^{-16}$)
100	1.90×10^{-15} ($\pm 0.95 \times 10^{-16}$)
700 H	2.47×10^{-15} ($\pm 2.5 \times 10^{-16}$)
100 H	1.72×10^{-15} ($\pm 1.7 \times 10^{-16}$)
600 C	1.96×10^{-15} ($\pm 2.0 \times 10^{-16}$)
100 C	2.27×10^{-15} ($\pm 2.3 \times 10^{-16}$)

Table 5.2: Neutron cross section without and with heater (H) and cooler (C)

In Figure 5.2 there are depicted the data written in the first part of Table 5.2 as function of designs frequency. In Figure 5.3 there are depicted all the data reported in Table 5.2 as function of temperature. The data are divided into three groups: in the first one (black squares) there are the data from the runs without any addition; in the second one (grey triangles) the data are about cooler; the last (grey circles) is about the runs with the heater.

The peak in the cross section trend is reached at 600 MHz (62 °C) with an increase in cross section of around 22% with respect the 100 MHz case, while the lower cross section is at 700 MHz (−23% with respect 600 MHz and −7% with respect 100 MHz). The cross section increase with temperature and then its suddenly decrease is in accordance with data already reported in [7].

As already explained, Figure 5.3 reports also a combination of experimental results obtained varying the operating frequencies and artificially varying the temperature with an external heater or cooler. As it can be noticed, data obtained with heather or cooler matches the one obtained with

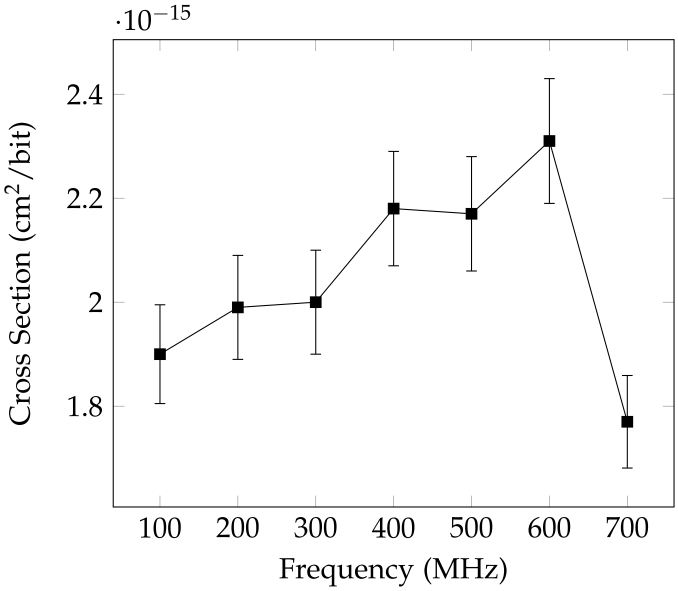


Figure 5.2: Cross section as function of frequency in our designs

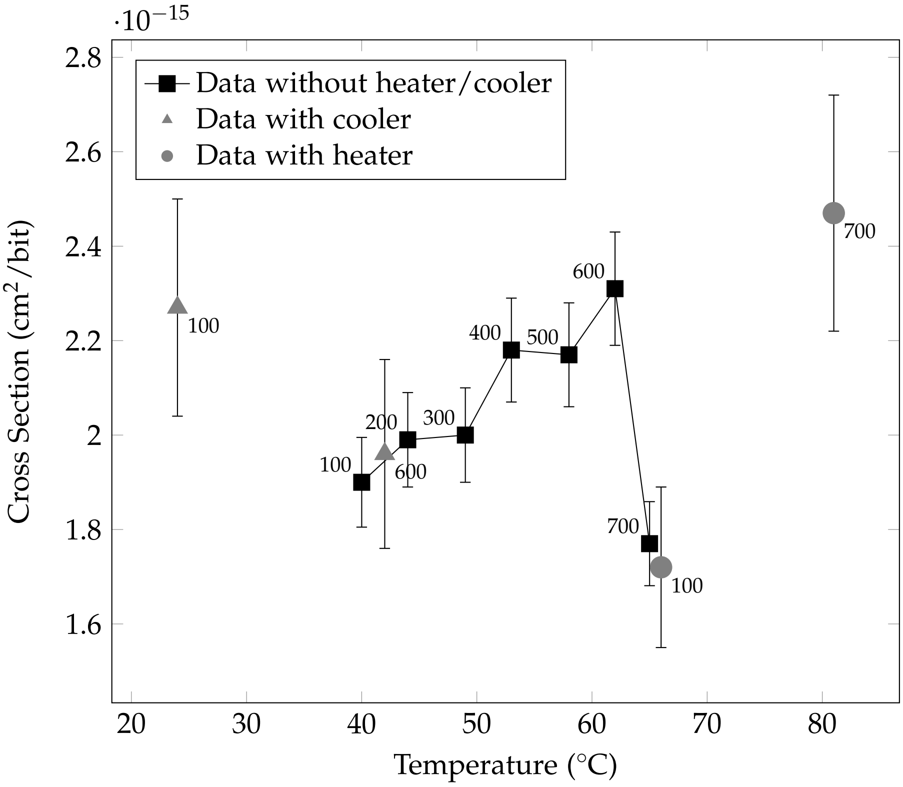


Figure 5.3: Cross section as function of temperature in our designs. The numbers near each point are the frequencies of running designs.

frequencies variations. For example, the device running at 200 MHz and 44 °C (without cooling) and at 600 MHz and 42 °C (with cooling) showed almost the same cross section, with only 1.4% difference. This is a strong evidence that the effects measured are indeed related to the temperature and not to the operating frequency. A similar situation is observed when running at 100 MHz and 66 °C (with the heater) and at 700 MHz and 65 °C. The extreme cases obtained with heating and cooling (81 °C and 24 °C) cannot be reproduced without these auxiliary devices. But it is interesting to observe that the optimum spot found round 65 °C is lost when the system reaches 81 °C. Also cooling to lower temperature (24 °C) did not bring benefits.

Cross Sections Comparison with Xilinx® Data

In [15] there are reported many data about the reliability of Xilinx® FPGAs, included the neutron cross section. The experiments for determining the latter were performed at *Los Alamos Neutron Science Center (LANSCE)* facility and the neutron cross section found for 7 series FPGAs is $6.99 \times 10^{-15}\,\mathrm{cm}^2/\mathrm{bit}$.

6. Conclusions

We have shown how the heat influences the error rate in an SRAM memory. A theoretical approach to this phenomenon is extremely difficult: in this process a great number of physical factors intervenes, as shown in Section 3.4, and these factors interact in a very complex way, since they are modified in different manners by the temperature at the same time.

The experiment described in this work has given a trend which can not be explained in an easy way. We have noted a rise in the cross section not for all temperatures, but only in the range 40-62 °C. A similar trend is depicted in Figure 3.6 [7]. Although this trend is difficult to explain, we have shown an important fact: the SEU rate depends on the device temperature and not only on the device frequency. Looking to the data obtained in cooler/heater sessions, we can see that they are compatible with the data obtained in sessions without any external temperature alteration.

Bibliography

[1] G. Furano. *Issues in (very) rad hard systems: an ESA perspective on use of COTS and space grade in JUICE mission*. ESA, 2011.

[2] Kenneth A. LaBel and Michael J. Sampson. *Issues in (very) rad hard systems: an ESA perspective on use of COTS and space grade in JUICE mission*. NASA, 2013.

[3] Kaushik Roy, Saibal Mukhopadhyay, and Hamid Mahmoodi-Meimand. "*Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits*". In: *Proceedings of the IEEE* (2003), pp. 305–327.

[4] Robert C. Baumann. "*Landmarks in Terrestrial Single-Event Effects*". In: *IEEE NSREC Short Course* Section III (2013).

[5] J. R. Srour, Cheryl J. Marshall, and Paul W. Marshall Marshall. "*Review of Displacement Damage Effects in Silicon Devices*". In: *IEEE Transactions on Nuclear Science* 50 (2003), pp. 653–670.

[6] Fernanda Lima Kastensmidt. "*SEE Mitigation Strategies for Digital Circuit Design Applicable to ASIC and FPGAs*". In: *IEEE NSREC Short Course* Section II (2007).

[7] M. Bagatin et al. "*Temperature Dependence of Netron-Induced Soft Errors in SRAMs*". In: *Microelectronics Reliability* (2012), pp. 289–293.

[8] *ZedBoard Hardware User's Guide*. Version 1.9. Avnet®. 2013. URL: `http://www.zedboard.org/documentation/1521`.

[9] *Zynq-7000 All Programmable SoC - Overview (DS190)*. Version 1.5. Xilinx®. 2013.

[10] *Zynq-7000 All Programmable SoC - Technical Reference Manual (UG585)*. Version 1.6. Xilinx®. 2013.

[11] *Zynq-7000 All Programmable SoC (XC7Z010 and XC7Z020): DC and AC Switching Characteristics (DS187)*. Version 1.7. Xilinx®. 2013.

[12] Markus Happe et al. "*Eight Ways to put your FPGA on Fire - A Systematic Study of Heat Generators*". In: *International Conference on Reconfigurable Computing and FPGAs (ReConFig)* (2012), pp. 1–6.

[13] D. M. Rowe. *Thermoelectrics Handbook: Macro to Nano*. Taylor & Francis, 2006.

[14] M. Violante et al. "*A New Hardware/Software Platform and a New 1/E Neutron Source for Soft Error Studies: Testing FPGAs at the ISIS Facility*". In: *IEEE Transactions on Nuclear Science* 54 (2007), pp. 1184–1189.

[15] *Device Reliability Report* Fourth Quarter 2013 *(UG116)*. Version 9.7. Xilinx®. 2014.

Acknowledgement

Thanks to nobody[1].

[1]Except: my family (Dario, Anna Maria, Ester, Francesco), all relatives (especially *'a me santula* Cesarina), Prof. Paccagnella, Prof. Rech, Prof. Kanstensmidt, Prof. Reis, Carlo, Cini, Cristian, Cristina, Ale Riccio Barba Chubby, Panch, Fox, Tognon, Momolo, Bersan called The God and The Son and The Holy Spirit and Holy Mary and Holy Pope, Bulbasaur, Capo, Rigatoma, Franz, Scarpa, Graspa, Saverio, Berni, Papa, the beautiful Vanin, the beautiful Rocco, the beautiful Bessegato, Biadene, Mattia, Matteo, Michael, Carraro, Massanzago, Anita, Martina, Francesca Ferrari, 8 people, maybe 9, Federico Gold Faoro, Andrea Pizzolato, Marcello Pavan, Daniele Clara Bertaso, Massimiliano Dalla Mura and his female friends, Nicolò Wiwa Rivato, Filippo Maker Faccio, Andrea Cecco Cecconato, GabryLele Roberti, Jorge Tonfat, Roberto Montoya, Lucas Tambara, Ronaldo Ferreira, Gabriel Nazar, Thiago Santini, *a segunda mae* Natalia Kornijezuk, Nathalia Matychevicz, Lis Mauri, Carol Schott, Cristina Tuzzin, Renato Navas, Gustavo Maia, Bruno Campelo, Tiago Castro, Thanise Füller e outras pessoas brasileiras, Tubo, Brundo, Tavarish Bonetto, Alessandro Renier, Arone, Tullio, Torelli Sudati Rugby Club, Quelli del Lunedì, the dirty communists of Lotta Comunista, the dirty Karl Marx, his friend Engels and all workers, Monty Python, Stanlio e Ollio, Fantozzi, The Flying Spaghetti Monster, Martinotti, Charmat, Paganini, Bach, Beethoven, Dvořák, Rossini, Vivaldi, Mussorgsky, Satriani, Vai, Malmsteen, Dio, Judas Priest, IEEE, IEEE Solid-State Circuits Society, IBM, Intel, TSMC, Rabaey, Chandrakasan, Maxwell, Boltzmann, Fermi, Dirac, Erdos, Shannon, Shockley, Bohr, Hawking, Von Neumann, Hilbert, Einstein, Heisenberg, Galileo, Gauss, Riemann, Lobačevskij, Aristarco, Röntgen, Ippocrate, Archimede, Nyquist, Newton, Arrhenius.
And other people.