

Diversity TMR: Proof of Concept in a Mixed-Signal Case

Gabriel de M. Borges, Luiz F. Gonçalves, Tiago R. Balen, Marcelo Lubaszewski

Universidade Federal do Rio Grande do Sul

Departamento de Engenharia Elétrica

Porto Alegre, RS, Brazil

{gabriel.borges, tiago.balen}@ufrgs.br, {luizfg, luba}@ece.ufrgs.br

Abstract— In this paper a design diversity fault tolerance technique is applied to a mixed-signal (MS) system. Three different implementations of a second order low-pass filter (which perform the same transfer function) associated to a majority voter are used to build the TMR scheme. The whole system is prototyped by using a programmable mixed-signal device. Some functional faults are injected into the circuit blocks and practical measurements are made on the prototyped system. Results show that the design diversity TMR is a feasible technique that can increase reliability of some classes of state-of-art MS circuits.

Keywords- fault tolerance; mixed-signal ; redundancy; design diversity;

I. INTRODUCTION

In applications requiring high reliability, such as avionics or medical systems, it is very important to reduce the probability of failure to the minimum possible. Besides the ability of a given circuit to detect a fault, it is desirable that the system maintains its normal operation under the occurrence of this fault or even be capable of correct it.

As the demand for smaller and low-power devices increases, state-of-art circuits are becoming more sensitive to the radiation effects (space applications), electromagnetic interference and other environmental interactions. Another concern is the natural ageing of the integrated circuits, which impacts in the system performance, availability and reliability.

Electronic systems employed in applications susceptible to radiation exposure may suffer long term effects know as TID (Total Ionizing Dose) and Single Event Effects (SEE), besides displacement damage and destructive effects [1]. A special class of SEE is the Single Event Upset (SEU) [2], which is an important matter of concern in systems with DRAM or SRAM memory blocks. In programmable circuits, such as the FPGAs, an error may occur, due to an SEU, when an energetic particle hits the programming SRAM blocks of the device, causing a bit-flip [3].

The effects of SEU in programmable analog technologies were first investigated in [4], showing that, the inversion of a single programming bit may be catastrophic to the system functionality.

Usually, the techniques used to add fault tolerance to electronic systems are based on hardware and/or time redundancy [5, 6, 7]. Among these techniques the most popular is the Triple Modular Redundancy (TMR). TMR consists in triplicating the designed circuit, in a way that the outputs of the circuit copies feed a majority voter. If there is an error in one of the triplicated blocks, two of them continue to operating properly and the correct value is chosen by the voter.

Traditional TMR techniques consider the replication of exact copies of the circuit to be protected [7, 8, 9]. However, if the copies are identical, environmental interactions, ageing or radiation incidence (in particular, Total Dose effects) may affect different circuit parts at same time, causing multiple faults. This way, errors may occur in more than one system copy output, and the voter may select faulty signals as system output.

Conversely, if the circuit copies in TMR systems are built with different architectures, the probability of multiple faults affecting different TMR blocks can be reduced, since each system copy may have different levels of resilience associated to the diverse fault generation mechanisms and sources.

The present work addresses the application of the concept of fault tolerance by design diversity [10] to mixed-signal technologies. The proposed scheme consists in a TMR system in which the design of interest is a low-pass filter.

The validation of the proposed scheme is made through the physical implementation of the system in a single mixed-signal programmable device, the PSoC (Programmable System-on-Chip) CY8C29466, from Cypress Semiconductor [11].

The rest of the paper is organized as follows: in Section II the diversity TMR technique is introduced. Section III presents the three different filter architectures considered in this case study. Section IV shows experimental results and section V concludes the work.

II. DIVERSITY TMR: THE CONCEPT

TMR is a well-known technique frequently applied to modern digital systems which require a high degree of reliability.

If multiple faults occur sequentially, with a relatively long inter-occurrence interval, the TMR system can tolerate it.

However, multiple faults occurring in more than one copy at the same time may cause the voter to take the wrong decision, if two or more TMR blocks present bit inversion in their outputs [12, 13, 14]. Faults generated by environmental interactions, ageing or radiation incidence may affect different circuit parts at same time, and are potential causes of multiple faults.

The use of different technologies or architectures to implement a triple modular redundancy scheme is an alternative way to increase the reliability of a system. This technique is referred to as “TMR based on design diversity” [15] and from here on it will be denoted by DTMR.

In the DTMR the system is replicated three times and their outputs are delivered to a voter. However, to achieve the design diversity, the three circuit copies are implemented with different technologies, algorithms or architectures.

In this work the DTMR is applied considering also different domains (analog and digital), and system levels (software and hardware), to build the system copies, as depicted in Figure 1.

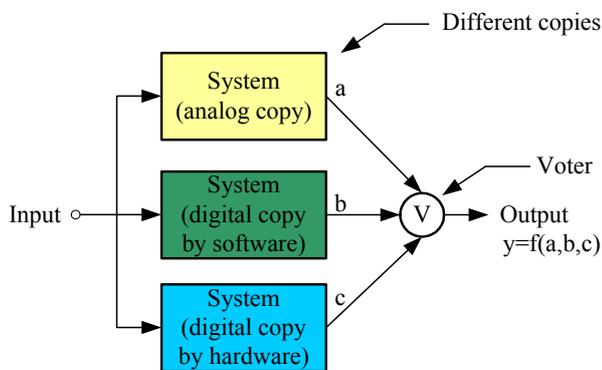


Figure 1. Mixed-Signal DTMR architecture.

Despite triple modular redundancy be simple and easy to implement, it implies in large area overhead and increase in the power consumption [9, 12, 13, 16]. Obviously, this new DTMR technique also leads to a significant area overhead. However, if reliability is the major concern, this aspect becomes less important due to the increasing of integration capacity of current microelectronics technologies.

III. MIXED-SIGNAL CASE STUDY

The design of interest considered in this work is a second order low-pass filter. The three copies of the filter perform the same function, but are implemented in three different architectures and domains, as follows:

- 1) An analog filter,
- 2) A digital filter implemented by software and
- 3) A digital filter implemented by hardware.

Furthermore, a voting system is implemented by software.

The considered analog filter is a Butterworth implementation with cutoff frequency of 450Hz. The digital filters (hardware and software) are designed to present the same frequency response of the analog block.

The whole system is prototyped in the PSoC CY8C29466, from Cypress Semiconductor [11], a mixed-signal programmable SoC. This component consists in a full mixed-signal platform with a microcontroller, memory blocks, data converters, besides a set of analog and digital user programmable analog and digital blocks [11].

The PSoC programmable digital blocks can be configured as counters or PWMs, for example. The analog blocks are used to implement analog filters, comparators and programmable gain amplifiers, among other several analog functions. In addition, the PSoC comprises special system resources such as multiply-accumulate blocks (MACs) and decimators, which provide additional mixed-signal capabilities.

A. Analog low-pass filter

The analog filter was implemented by using the PSoC programming library, in which a pre-built low-pass filter is available. This implementation uses the switched-capacitor programmable blocks of the device.

The corner frequency and quality factor are functions of the capacitances and switching frequency of the programmable capacitors. A proprietary tool, called *PSoC Designer* [11], is used to program the component. In addition to specify the parameters of filters, one must allocate it properly in the programmable analog/MS programmable array. Figure 2 shows the programmed functions and their placement in the programmable array. As can be seen in Figure 2, the analog filter is allocated in the left blocks of the analog/MS array.

The PGA2 block in Figure 2 is a Programmable Gain Amplifier, and it is programmed with unitary gain to make a connection of the filter with the *Port[0]1* I/O, which is the filter external input.

B. Digital low-pass filter by software

The process to implement a digital filter includes: analog-to-digital conversion, digital signal processing and digital-to-analog conversion. The digital signal processing is implemented in software, i.e., the calculations are made by using the Arithmetic Logic Unit (ALU) from the PSoC microprocessor.

The A/D converter is a 7 to 13-Bit *Variable Resolution Incremental ADC*. The sample rate is defined by the data clock divided by the converter integration time plus the time it takes to perform the result calculations (*CalcTime*). The integrating time is the period when the input signal is being sampled by the ADC. In this implementation the *DataClock* is 3MHz and the *CalcTime* is 50 cycles. Thus, with a 7-bit resolution for the ADC, the sample rate can be calculated as follows:

$$SampleRate = \frac{DataClock}{2^{Bits+2} + CalcTime} = 5338 \text{ Hz} \quad (1)$$

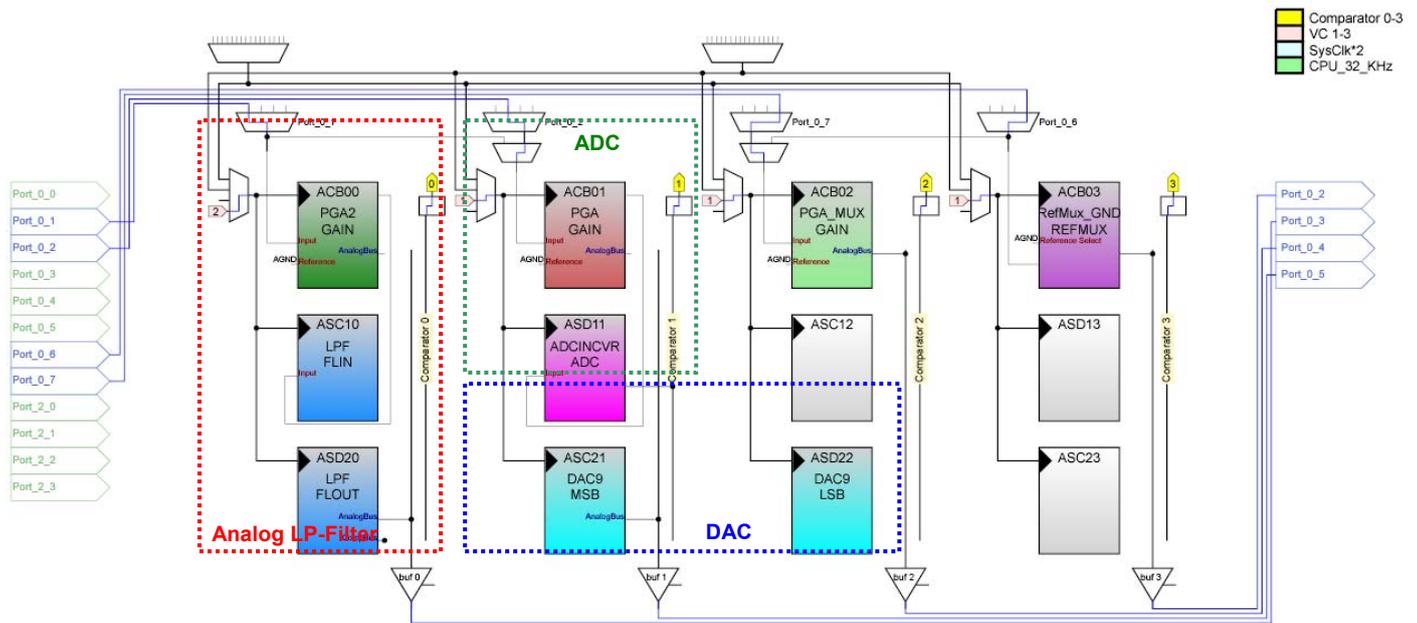


Figure 2. Allocation of selected user modules in the PSoC array.

The digital filter implemented is a second order FIR filter with integer coefficients, and its transfer function in the discrete-time domain is given by:

$$y = 4x[0] + 7x[1] + 4x[2] \quad (2)$$

where the terms 4, 7 and 4 are the filter coefficients; $x[0]$, $x[1]$ and $x[2]$ are the signal samples, with the respective delays, and y is the filter's digital output. Figure 3 shows part of the program code which implements the digital signal processing, where the command $x[0]=ADCINCVR_iGetDataClearFlag()$ is used to get a new sample from the ADC.

```

int a[3]={4,7,4};
...
for (count=3; count>0; count--)
    x[count]=x[count-1];
x[0]=ADCINCVR_iGetDataClearFlag();
...
for (i=0;i<3;i++)
    software+=a[i]*x[i];

```

Figure 3. Algorithm of the digital filter by software.

After this computation process, the samples are sent to a D/A converter, programmed with 9-bit resolution. The DAC is allocated in two mixed-signal programmable blocks (Figure 2, bottom).

C. Digital low-pass filter by hardware

In order to implement the hardware digital filter, an additional system resource of the PSoC device, called MAC (multiply-accumulate unit) is used. The MAC is a dedicated

logic circuit used in digital signal processing (DSP) devices that implements a multiplier followed by an adder and an accumulator register, which stores the result.

Therefore, since the MAC unit performs the operations of multiplication, addition and accumulation without using the microprocessor, this resource was chosen to implement the third copy of the filter, in order to implement the Diversity TMR technique. This hardware-based digital filter presents the same number of taps and coefficient values of its software-based copy.

The MAC unit presents three main data registers. The signal samples from the ADC are stored in the MUL_X register and each filter coefficient is stored in the MAC_Y register. Then, the MAC unit automatically performs the multiplication and stores the product into the ACC_DR0 register. It is necessary to include some code lines in the main program to configure and preset the MAC registers, as shown in Figure 4.

```

MAC_CL0=0;
...
MUL_X=x[0]; MAC_Y=4;
MUL_X=x[1]; MAC_Y=7;
MUL_X=x[2]; MAC_Y=4;
hardware=ACC_DR0;

```

Figure 4. Configuration code of the digital filter by hardware.

The product of the next multiplication is added to the previous product and accumulated in the ACC_DR0. When the three coefficients are multiplied with three samples the value in ACC_DR0 is sent to the D/A converter and the MAC_CL0 register is cleared in order to perform the next multiply-accumulate cycle.

D. The voter

An efficient voting mechanism for digital signals is designed to select as system output the value that is the majority among the redundant outputs [17].

The extension of TMR concept to the analog case requires the design and implementation of a voting scheme suitable for dealing with analog signals.

The voting scheme adopted in this work consists in three comparators plus a decision element, as seen in Figure 5. The comparators performs mutual comparisons between the outputs of the three TMR blocks (comprising all possible combinations) generating the selection signals for the decision element. The comparisons are actually subtractions between each signal sample generated by the filters, producing three error signals (e_1 , e_2 and e_3). Based on the error signals the decision element evaluates the correctness of the signals V_1 , V_2 and V_3 , and selects one correct signal to be presented at the system global output.

Given to the fact that two of the filter copies are implemented in the digital domain, the voter was chosen to be implemented in software. For this reason, an A/D converter must be employed to digitize the signal at the output of the analog filter, in a way that the comparison can be performed in the digital domain. Obviously, digital filters naturally process digital values of the sampled signal.

This way, the signal samples at the outputs of the digital filters are directly delivered to the voting element.

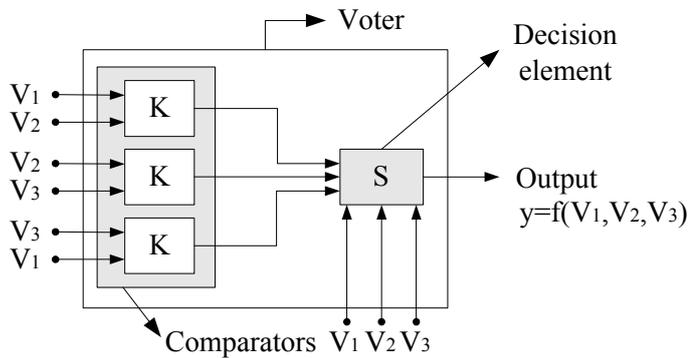


Figure 5. Voting scheme concept.

Despite the logic of voting system be fully digital, one must take into account that the digital samples are originated from analog signals, which, in turn, are not necessarily equal. Therefore, some level of difference may be acceptable and a tolerance window for the difference between the signals must be considered [4, 17].

Taking this into account, if the difference between two samples (from different blocks) exceeds a certain pre-defined tolerance, the voter may detect an error. If this error affects only one of the filters, the decision element is able to identify the correct blocks selecting one of them to be switched to the

system output. The part of the program which implements the voter algorithm is shown in Figure 6.

```

e1=abs(V1-V2); // Subtractions of
e2=abs(V2-V3); // digitized signals
e3=abs(V3-V1); // of the 3 filters

if (e1<=2 || e3<=2)
{
MUX_OUT_InputSelect(MUX_OUT_PORT0_3);
}
if (e1<=2 || e2<=2)
{
MUX_OUT_InputSelect(MUX_OUT_PORT0_5);
DAC9_WriteStall(software);
}
if (e2<=2 || e3<=2)
{
MUX_OUT_InputSelect(MUX_OUT_PORT0_5);
DAC9_WriteStall(hardware);
}

```

Figure 6. Voter algorithm.

As can be seen in the program block shown in Figure 6, a multiplexer is used to select one of the filter outputs according to the results of the subtractions (MUX_OUT_PORT0_3 selects the analog filter, while MUX_OUT_PORT0_5 selects a digital filter).

IV. EXPERIMENTAL RESULTS

Some practical measurements were performed in order to verify the functionality of this MS-DTMR system. In these experiments functional deviations were injected into each filter, while the signals at the system outputs were monitored.

Figure 7, 8 and 9 shows the acquired waveforms. The acquired data were plotted by using the software MATLAB, with which the white noise was removed from all samples for a better visualization.

At this time single functional deviations were injected. In the digital filter by hardware an error was introduced by changing a coefficient of the transfer function of the filter. For the digital filter by software a coefficient value shift was also considered, while a modification in a capacitor value was injected in the analog SC filter.

Functional deviations in the digital filters may be due to bit-inversions in the binary values of their coefficients, as well as, in the digital signal samples being processed (besides, of course, due to errors in the control part of the processor).

Similarly, since the values of capacitors in the analog module are programmed by switching weighted groups of unit capacitors (programming capacitor banks), and the state of the programming switches is defined trough the programming memory, it is possible that faults in the analog module are also caused by bit inversions.

In applications in which the circuit is exposed to radiation environments SEU faults may be the cause of these bit-flips, which, in turn, may lead to functional deviations.

As can be seen in each case (Figures 7, 8 and 9), if one of the three filters presents a fault the others are kept in normal operation, in a way that the voter is able to select one of the two correct signals. When the faulty module is one of the digital filters (hardware or software) the signal selected by the voter is the output of the analog filter (Figures 7 and 8), since this selection is made in the first “if” block of the voter algorithm.

Similarly, when the analog filter presents a deviation related to their digital copies the output of the filter implemented by software is chosen by the voter as the system output (Figure 9).

In these experiments the tolerance window considered for the error signals is 2 LSBs, as can be seen in Figure 6. The suitable values of this window should be evaluated by the designer, since different applications and required reliability levels may demand different tolerance windows.

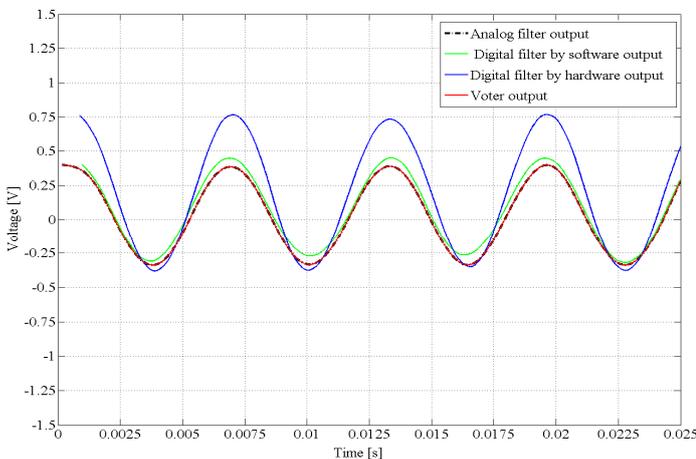


Figure 7. Outputs of the DTMR blocks and voter with a functional deviation in the digital filter by hardware

V. CONCLUSIONS

In this work the concept of redundancy with diversity was applied to a mixed-signal design as a proposal to increase its fault tolerance characteristic. In this preliminary analysis the effectiveness of the scheme to cope with single functional deviations in one of the system modules was investigated.

The DTMR system was prototyped in a single commercial programmable device. The system consists in three different versions of a low-pass filter (analog, digital by software and digital by hardware), in addition to a voter system.

Results indicate that the system is able to detect single functional deviations, just like the traditional TMR. However, the design diversity increases the degree of reliability, since

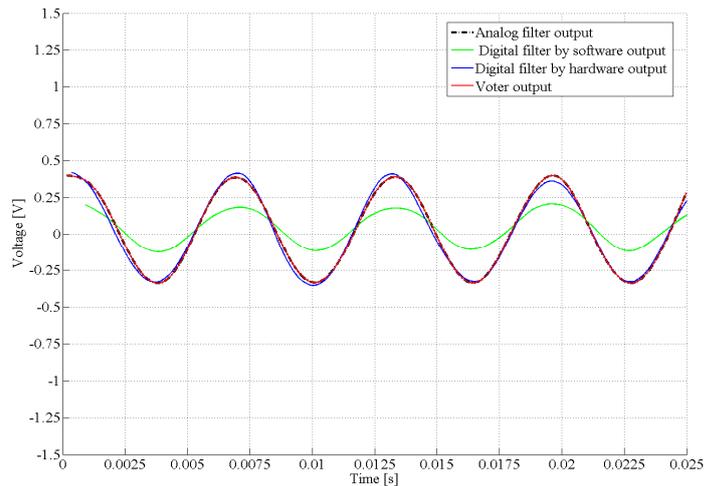


Figure 8. Outputs of DTMR blocks and voter with a functional deviation in the digital filter by software

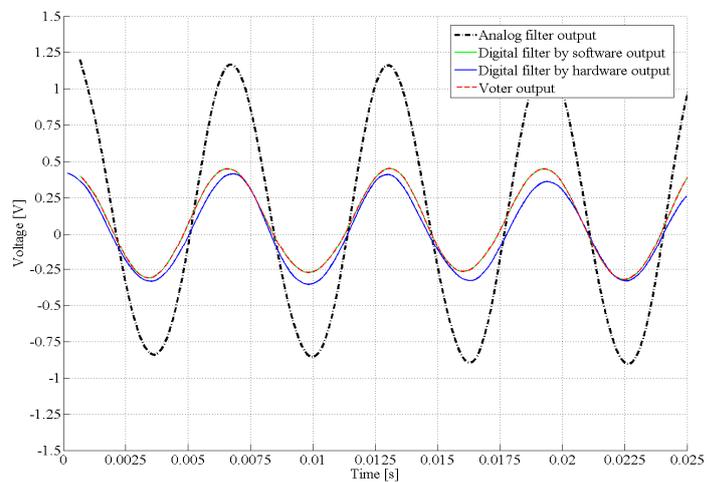


Figure 9. Outputs of DTMR blocks and voter with a functional deviation in the analog filter.

each redundant module may have different levels of resilience associated to the diverse fault generation mechanisms and sources.

Considering the limitations of the programmable MS device used in this case study, the system presents some weak points, such as the sharing of the ADC and DAC by the digital filters, as well as the voter system, which is dependent of the correct functioning of the microprocessor. However, this case study consists in an initial study of the DTMR concept proposed in this work and shows that it is potentially applicable to complex systems supported by state-of-art technologies.

Current activities of this research are focused on performing a fault injection campaign in order to obtain a quantitative evaluation regarding the effectiveness of the

system to deal with single and double faults. Additionally, the extension of this technique to other types of analog functions is being studied.

VI. ACKNOWLEDGEMENTS

This research is supported by the CNPq Brazilian Research Agency by contract number 180115/2009-0.

VII. REFERENCES

- [1] Velasco, R.; Fouillat, P.; Reis, R. (Org.). "Radiation Effects on Embedded Systems". Springer, 2007.
- [2] Wang, F.; Agrawal, V. D. "Single Event Upset: An Embedded Tutorial". IEEE 21st International Conference on VLSI Design, 2008.
- [3] Kastensmidt, F. L.; Neuberger, Gustavo; Carro, Luigi ; Rei, Ricardo ; Hentschke, R. . Designing Fault-tolerant Techniques for SRAM-based FPGAs. IEEE Design and Test of Computers, USA, v. 21, n. 6, p. 552-562, 2004.
- [4] T. R. Balen, F. Leite, F. L. Kastensmidt, M. S. Lubaszewski, "A Self-Checking Scheme to Mitigate Single Event Upset Effects in SRAM-Based FPGAs", *Transactions on Nuclear Science*, IEEE, vol. 56, n. 4, Aug. 2009, p. 1151-1162.
- [5] J. Vial, A. Bosio, P. Girard, C. Landrault, S. Pravossoudovitch, A. Virazel, "Using TMR Architectures for Yield Improvement". International Symposium on Defect and Fault Tolerance of VLSI Systems, Oct. 2008, p. 7-15.
- [6] K. S. Morgan, D. L. McMurtrey, B. H. Pratt, M. J. Wirthlin, "A Comparison of TMR with Alternative Fault Tolerant Design Techniques for FPGAs". *Transactions on Nuclear Science*, IEEE, vol. 54, n. 6, Dec. 2007, p. 2065 – 2072.
- [7] A. Anghel, D. Alexandrescu, M Nicolaidis. "Evaluation of a Soft Error Tolerance technique based on Time and or Hardware Redundancy". Proc. of IEEE Integrated Circuits and Systems Design, pp. 237-242, Sept. 2000.
- [8] E. Elnozahy, R. Melhem, D. Mosse, "Energy-efficient Duplex and TMR Real-time Systems". 23rd Real-Time Systems Symposium, IEEE, Dec. 2002, p. 256-266.
- [9] F. Lima, C. Carmichael, J. Fabula, R. Padovani, R. Reis, "A Fault Injection Analysis of Virtex FPGA TMR Design Methodology", 6th European Conference on Radiations and Its Effects on Components and Systems, IEEE, Grenoble, France, 10-14 Sept. 2001, p. 275-282.
- [10] Avizienis, A.; Kelly, J. P. J. "Fault tolerance by design diversity - Concepts and experiments". Computer. Vol. 17, pp. 67-80. Aug. 1984.
- [11] Cypress Microsystems. CY8C2XXXX Family Datasheet, 2002. Available in www.cypress.com.
- [12] H. Kim, K. G. Shin, "Design and Analysis of an Optimal Instruction-retry Policy for TMR Controller Computers", *Transactions on Computers*, IEEE, vol. 45, n. 11, Nov. 1996, p. 1217-1225.
- [13] M. Radu, D. Pitica, C. Posteuca, "Reliability and Failure Analysis of Voting Circuits in Hardware Redundant Design", International Symposium on Electronic Materials and Packaging, 2000, p. 421-423.
- [14] J. Yoon, H. Kim, "Time-redundant Recovery Policy of TMR Failures Using Rollback and Roll-forward Methods", *Proceedings on Computers and Digital Techniques*, IEEE, vol. 147, n. 2, Mar, 2000, p. 124-132.
- [15] Mitra, S.; Saxena, N.R.; McCluskey, E.J.; "A design diversity metric and reliability analysis for redundant systems". Proceedings of International Test Conference - 1999, p. 662-671.
- [16] B. Shim, N. R. Shanbhag, "Energy Efficient Soft Error-tolerant Digital Signal Processing", *Transactions on Very Large Scale Integration (VLSI) Systems*, IEEE, vol. 14, n. 4, Apr. 2006, p. 336-348.
- [17] D. V. G Vega, "Diseño para Testabilidad y Tolerancia a Fallos en Circuitos Analógicos". Thesis (doctorate in physics), Sevilla University, 1995.