An Accurate Single Event Effect Digital Design Flow for Reliable System Level Design

Julian Pontes and Ney Calazans
Faculty of Informatics - FACIN, - PUCRS
Porto Alegre, RS, Brazil
{julian.pontes, ney.calazans}@pucrs.br

Pascal Vivet
CEA-LETI
Grenoble, France
pascal.vivet@cea.fr

***Abstract*— Similar to local variations and signal integrity problems, Single Event Effects (SEEs) are a new design concern for digital system design that arises in deep sub-micron technologies. In order to design reliable digital systems in such technologies, it is mandatory to precisely model and take into account SEEs. This paper proposes a new accurate design flow to model non-permanent SEE effects that can be applied at system level for reliable digital circuit design. Starting from low level SPICE-accurate simulations, SEEs are characterized, modeled and simulated in the digital design using commercial and well accepted standards and tools. The proposed design flow has been fully validated through a complete digital design, a cryptographic core implemented in a 32nm CMOS technology. Finally, using the SEE design flow, the paper presents some reliability impact analysis, both at standard cell level and design level.**

Keywords-component; Single event effects, soft errors, radiation hardening.

I. INTRODUCTION

The shrinking transistor and wiring dimensions of deep submicron technologies bring additional constraints to traditional digital system design. The effect of local variations, the increasing influence of signal integrity problems, and the phenomena related to high energy particle impacts on silicon structures are examples of these new types of constraints [1]. Signal integrity and process variations already count with some well established solutions available in commercial CAD tools and in this way can somehow be treated at design time. The same is not true for effects caused by high energy particle impacts on silicon. A high energy particle or ion can cause different types of effects when crossing silicon structures. The resultant effect depends on the linear energy transfer (LET) of the ion and on the state of the crossed junction (if any) [2]. When the victim junction is in reverse bias, the generated electron-hole pairs produced by the ion are collected by the electrical field of the reversed junction. The magnitude of the current induced by the ion strike determines if the resulting effect will be permanent or non-permanent. A non-permanent event is classified as a Single Event Transient (SET), if it only creates a glitch in combinational logic, or a Single Event Upset (SEU), when the glitch is stored by a memory element. An SEU is usually referred as a Soft Error in the literature. Predictions exist that on advanced nodes technologies, high Single Event Rates (SERs) will become significant robustness problems for Systems on Chip (SoCs) [3] [4]. In a digital circuit, the SER is measured in failures-in-time FIT (1 FIT = 1 failure in one billion device hours) [5]. Several techniques and methods are available in the literature to increase the reliability of digital circuits by reducing the SER [3]. These methods usually incur in area and power overheads, and in a reduction in system performance. To satisfy robustness requirements without incurring in wasted resources and/or design time delays, it is necessary to develop methods and tools to evaluate in an accurate and fast way the robustness of digital system implementations. The flow presented in this work allows the use of digital level simulation by digital designers to meet robustness specifications for new generations of SoCs.

The rest of this work is divided in five sections. Section II presents works related to proposition of this paper. Section III introduces the proposed flow, while Section IV shows the flow implementation details using commercial CAD and standards. Section V discusses results achieved with the flow when applied to an RSA cryptographic core design. Section VI ends the paper with some conclusions and directions for future work.

II. RELATED WORKS

There are several SEE models available in the literature. These models cover different abstraction levels [4]. A first abstraction level consists in measuring the injection of charge in silicon structures using laser sources, protons or ion radiation [6] [7]. At this level, accurate values for collected charge and current shape are extracted. The method is well suited to characterize fabrication technology processes, memory elements or small test circuits. The characterization of a complex digital circuit at this level is unfeasible, since it must include the measurement of every node at different circuit states and PVT conditions. At the same level of abstraction are the models obtained using physical simulators. These simulators model the (physical) interactions between radiation and materials. Clearly, the same restrictions found at the physical measurement level apply to physical simulators to obtain system level validation for SEEs. Nonetheless, measurements obtained at these two levels are extremely useful as input parameters for the next abstraction level: electrical simulations.

Electrical simulations are faster if compared to their physical counterparts. At this level, however, the silicon

978-3-9810801-8-6/DATE12/©2012 EDAA

irradiation process is not directly modeled. The process is usually captured using an exponential current source as Equation (1) [4] [8].

$$I(t) = I_0(e^{-t/\tau\alpha} - e^{-t/\tau\beta}) \quad (1)$$

Parameters τα and τβ of the exponential source are provided at the physical modeling level - through either measurements or simulations - and represent the behavior of the silicon charge collection. These parameters are highly dependent on the technology. Employing electrical simulations, it is possible to extract the behavior of a circuit when submitted to a specific exponential current curve. Two measurements are used to characterize a digital circuit when facing a charge impact: the critical charge and the output pulse width. Critical charge is the minimum amount of charge necessary to create an event at the output of a digital circuit. The pulse width can be just the size of the pulse or in some works its shape as well [5] [9]. Both, critical charge and pulse width are dependent on the circuit state and on the node in which the exponential source is applied. Even if it is faster when compared to physical abstraction levels, electrical simulations are not adequate for analyzing the overall SEE behavior in complex digital systems. To be able to tackle the SEE in complex digital systems it is fundamental to treat such events at higher abstraction levels. This next abstraction is the digital design level, which models SEEs as digital pulses [8]. Measurements obtained here relate to the Single Event Rate. The SER in digital circuits can be statistically modeled by Equation (2) [3] [4].

$$SER(n_i) = P_{SEE}(n_i) \times P_{sensitized}(n_i) \times P_{latched}(n_i) \quad (2)$$

Here:

- P_{SEE} is the rate at which SEEs occur in a circuit node ni with sufficient strength to propagate to a memory element;
- $P_{sensitized}$ is the probability that a functionally sensitized path from node ni to a memory element exist;
- $P_{latched}$ is the probability that the SEE be captured by a memory element.

Equation (2) represents the relationship among the cell electrical filtering property (P_{SEE}), the circuit logical filtering property ($P_{sensitized}$) and the memory elements' sampling filtering property ($P_{latched}$). The logical filtering and sampling window are easily verified using a digital simulator. Nevertheless, the electrical filtering property of SEEs in standard cells is not present in a digital simulation flow. In this way, the simple insertion of digital pulses is not enough to determine circuit robustness, since a pulse may have different meanings, depending on the node where the pulse appears and on the circuit state. Another way to estimate the SER is through analytical methods [3] [8]. In an analytical method it is possible to determine if a digital pulse produced in a specific node can reach the input of a register and consequently create an SEU. Analytical models are adequate for small size combinational circuits and in this way enable estimating the $P_{sensitized}$ part of Equation (2), but the time required to evaluate a complex circuit increases exponentially with circuit complexity [3]. The sequential behavior of a circuit is difficult to predict using this method, which makes it not suitable for predicting the SER in complex digital systems. In this sense, analytical models fail to determine the real failure rate at the boundary of a System on a Chip or of an IP Core. However, the precise definition of the system failure rate is a necessary measure to a complete SEE modeling.

III. SEE DIGITAL DESIGN FLOW

The previous Section showed that to address SEE-resistant SoC design it is mandatory to provide an accurate modeling of SEE faults at the digital level. This modeling has to be as accurate as possible compared to the electrical simulation level. With this goal, this work proposes the design flow of Figure 1. The flow is mostly based on existing tools and standard formats, for providing wide usability. Figure 1 comprises five columns, where the first, second and fifth columns present some new design steps to provide an SEE-aware design. The second and fourth columns represent a simplified conventional SoC design flow.

The SEE-aware flow was conceived to fit in a conventional digital design flow. The first step is the *SEE Characterization*, represented in the first column of Figure 1. This step performs SEE susceptibility and SEE timing measurements in the (standard) cell library. Characterization employs exponential current sources with parameters that extractable from measurement or simulations at the physical level. These parameters are represented as the *SEE Technology Model* in Figure 1. From the *SEE Characterization*, an encompassing set of electrical characteristics are extracted from electrical simulations, to form the *SEE Database*, including:

- Voltage level before SEE current insertion;
- Maximum Voltage reached at the victim node;
- Minimum Voltage reached at the victim node;
- SEE critical charge;
- Output pulse width.

Based on the result of the SEE characterization, the Liberty standard [11], used for timing and power description of the cells is adapted to hold SEE information about critical charge and pulse width in the *Post Processing* step. Adaptation of Liberty is achieved applying a transformation from critical charge to timing. Transformations permit a digital simulator to perform the electrical and logic filtering processes. In this way, the proposed may cover all the components of Equation (2) and thus provide accurate results related to the SER of a circuit.

To model the cells' SEE behavior, a new input pin is created. This pin is necessary to capture the effects of the SEE on the output pin of the cells and is added to each library cell. The new pin allows modeling both, the SEE behavior in cells and their timing properties. Figure 2 shows an example modeling scheme (for a 2-input NAND gate). The cell keeps the original behavior when the SEE pin is not asserted. In the presence of an SEE (SEE = '1'), the cell has its behavior changed, due to the bit flip caused by a radiation particle impact, for example. This simple change in the cell description enables the implementation of a flow capable to support SEE modeling in digital standard cells.

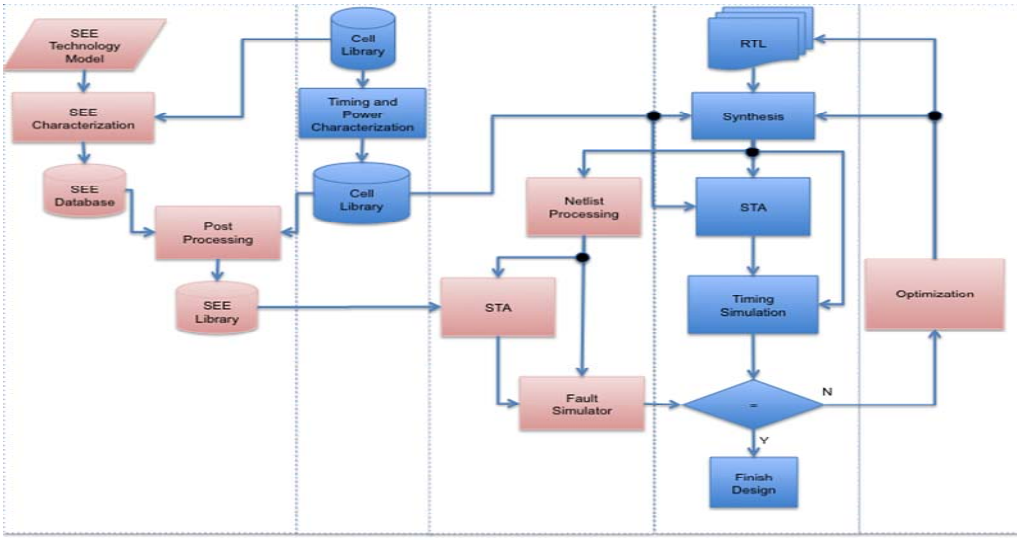
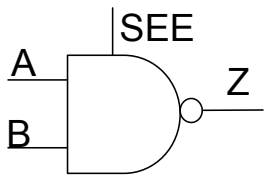


Figure 1- Proposed flow for digital circuit design based on SEE constraints.

Using this modification is the first step to create a verification environment (third column of Figure 1) for SEE where events can be created with controlled rates and duration in a *Fault Simulator*. However, this is not enough yet to permit SER measurements and analyses.



SEE	A	B	Z
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Figure 2 – Example of a modified NAND gate for SEE simulation.

Digital simulators implement timing filtering processes for glitches using a pulse control mechanism. In the default mode, a digital simulator exhibits the inertial delay mode for simulating the cell behavior in a digital circuit. In the inertial delay mode, any glitch in a cell is filtered if the glitch size is smaller than the propagation delay of the cell. In order to take advantage of this simulator property, a conversion from critical charge into delays is performed in the *Post Processing* step of the flow of Figure 1. In this way, the simulator is capable to filter SEEs indications with less charge than required for a cell in a given state. The charge to timing conversion uses Equations (3) and (4):

$$\text{SEE\_Charge} = \int_{t1}^{t2} I_0 (e^{-t/\tau\alpha} - e^{-t/\tau\beta}) \qquad (3)$$

$$\text{SEE\_Modeling\_Charge} = \int_{0}^{tx} I(t)\,. \qquad (4)$$

In Equation (4), I(t) is an equivalent constant current necessary to create the critical charge in a time interval defined as *tx*. The equivalent time (*tx*) is extracted from the relationship between charge and timing. Thus, *tx* is the value of the pulse width (measured in seconds) generated by a charge of one Coulomb. This charge to time transformation is applied as a characterization post processing phase, before entering the SEE effect within the timing library.

Finally, to analyze a complete system level design using this fault modeling at cell level, a static timing analysis tool (STA) generates all values of SEE timing. The timing of the cell is kept and a new timing arc is added from the SEE pin to the cell output. This process can be applied to either combinational and sequential cells with single or multiple outputs.

Characterization values are applied to instances in the netlist, including the timing arcs of the new input pin SEE to the cell output. Based on these values, it is possible to perform an SEE timing simulation on the design. This step is performed using a *fault simulator*. The fault simulator is responsible for generating the SEE indication in the design instances. SEE injection can be fully controlled by the fault simulator. It can control both, SEE injection rate and SEE injection charge. The fault simulator can verify if the desired levels of robustness are reached in a specific design implementation. Estimations generated by the fault simulator can be used to implement design *optimizations*.

IV. SEE Flow Implementation

A. *SEE Characterization*

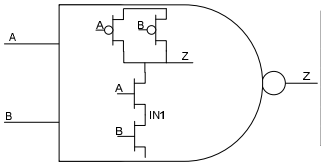
All current standard cell digital design flows work based on some estimation of cell electrical properties (timing, power). The SEE characterization is based on the same principle: extraction of the SEE susceptibility from electrical simulations under distinct PVT conditions. Cell characterization is achieved through electrical simulation. Since every cell state has a different level of susceptibility, characterization is conducted

for all steady states of a cell. For each steady state, just the nodes that have a reversed bias junction are considered. The transient states, those where one output of the cell is switching, are ignored, since the voltage in switching junctions is smaller than the supply voltage. Thus, they do not represent an SEE-worst case scenario. Figure 3 depicts the characterization method. The process referred as *FindCriticalCharge()* basically performs a binary search for the minimum charge of each susceptible node in a state.

```
SEE Characterization()
  for each cell in the library
    for each cell state
      Nodes = FindSusceptibibleNodes(cell)
      Charge(cell,state) = FindCriticalCharge(Nodes)
end
```

Figure 3 - Method for SEE characterization of standard cells.

Figure 4 shows an example SEE characterization. In this example, it is possible to compare the critical charge and pulse width for a 2-input NAND gate with two different drive strengths when driving a 5pF output load. For State=101, the critical node is the internal node IN1, while for the other states the critical node is the output pin. The values for critical charge where normalized based on the most robust combinational cell in the library.



Cell State			Normalized Critical Charge	
A	B	Z	NAND X3	NAND X20
0	0	1	0.0525	0.3055
0	1	1	0.0267	0.1529
1	0	1	0.0267	0.1535
1	1	0	0.00172	0.1033

Figure 4 - Characterization values of a 2-input NAND gate.

B. Fault Simulator

Fault simulation is performed using digital simulation with timing annotation. The principle of the SEE simulation is based on inertial delay properties. Figure 5 shows the adapted verification environment for SEE evaluation. SEE verification reuses the verification environment and test cases generated for the system behavior and timing verification. A fault generator module is introduced to produce the SEE indication in each instance of the design. The environment was designed using SystemC code. The process to select the victim instance is based on a random number generator, which picks a number in the interval between 0 and *n*, where *n* is the number of instances in the design. The fault generator is able to create single and multiple events. The event rate insertion can be easily controlled and allows generating events anywhere within a clock cycle.

C. Cell Verilog Behavior Model

To permit SEE generation and propagation, the SEE to Output pin behavioral and timing arc is created. To permit simulation of the new behavior as well as SEE and timing annotation, the changes are performed in the cell Verilog model.

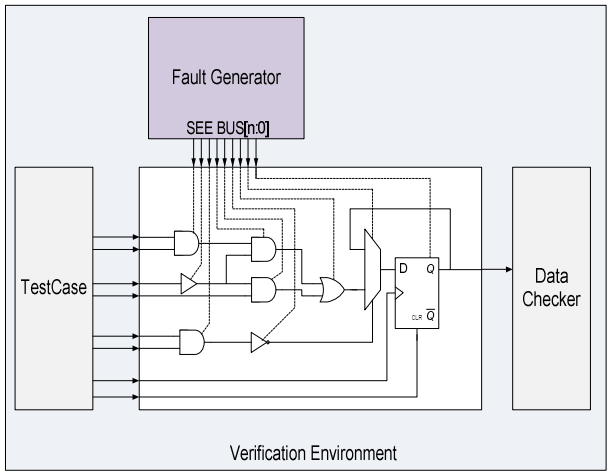


Figure 5- Fault simulation environment.

The left side of Figure 6 shows the *Verilog udp* model generated to implement the bit flip at the output of a digital cell. In this *udp* model, when the SEE input pin is equal to logic level '1', then the cell has its behavior altered. At the right side of Figure 6 the *SEE udp* is instantiated in a 2-input NAND gate Verilog model. In the cell Verilog model the timing arc is added between the input pin SEE and the output cell pin. This timing description is state dependent, to allow a better pulse description.

```
primitive u_seu (Z, A, SEE);

  output Z;
  input  A, SEE;

  table

  // A   SEE  :  Z
    0     0   :  0 ;
    1     0   :  1 ;
    0     1   :  1 ;
    1     1   :  0 ;
  endtable

endprimitive
```

```
module NAND2 (Z, A, B, SEE);
        output Z;
        input A;
        input B;
        input SEE;

        and   U1 (INTERNAL1, A, B) ;
        not   U2 (INTERNAL2, INTERNAL1) ;
        u_seu     (Z, INTERNAL2, SEE) ;

        specify
                if (B) (A -=> Z) = (0.01,0.01);
                if (A) (B -=> Z) = (0.01,0.01);
                if (!A && !B) (SEE => Z) = (0.01,0.01);
                if (!A && B) (SEE => Z) = (0.01,0.01);
                if (A && !B) (SEE => Z) = (0.01,0.01);
                if (A && B) (SEE => Z) = (0.01,0.01);
        endspecify
endmodule
```

Figure 6 – SEE adapted Verilog example for a 2-input NAND gate.

D. Cell Timing Liberty Model

The most accepted timing description standard for digital cells is the Liberty standard [11]. SEE susceptibility, after transformed in timing values, is modeled inside the Liberty format using the Non Linear Delay Model (NLDM). Using NLDM tables it is possible to represent the state dependence of the cell susceptibility as a function of the output load capacitance of the cell. Figure 7 shows the 2-input AND Liberty file on the left side and the corresponding Standard Delay Format (sdf) on the right side. The values in the Liberty files are used to guide the synthesis tools when searching the best cells to implement a given function, as well as to annotate the cell timing properties. In this way, values inside the Liberty file are used to overwrite the default values of the VHDL or Verilog models for timing simulation. This is usually done in

two steps: the cell delay file descriptor generation and the behavioral and timing model association.

```
cell(NAND2){
  pin(SEE){
   capacitance : 0;
   direction : input;
  }
 pin(Z){
   timing(){
     related_pin : "SEE";
     when : "A' * B";
     sdf_cond : "!A && B";
     timing_sense : negative_unate;
     cell_fall(SEU_table_9){
       values(" 1.10503 1.10565 … ");
     }
     cell_rise(SEU_table_9){
       values(" 0.00775537 0.0003982 …");
     }
     fall_transition(table_10){ values("0"); }
     rise_transition(table_10){ values("0"); }
   }
 }
```

```
(CELL
(CELLTYPE  "NAND2")
(INSTANCE  inst_multruegte_70_26trueg12644)
 (DELAY
       (ABSOLUTE
       (COND A&&!B (IOPATH SEE Z  (0.005::0.010) (1.112::1.113)))
       (COND !A&&B (IOPATH SEE Z  (0.005::0.004) (1.107::1.107)))
       (COND !A&&!B (IOPATH SEE Z  (0.005::0.007) (2.214::2.215)))
       (COND A&&B (IOPATH SEE Z  (0.751::0.751) (0.014::0.010)))
       (COND B (IOPATH A Z  (0.014::0.016) (0.019::0.022)))
       (COND A (IOPATH B Z  (0.016::0.020) (0.020::0.025)))
       )
  )
)
```

Figure 7 – Example of a timing library fragment for a 2-input NAND gate with timing arcs from SEE pin to output Z.

Figure 8 shows the waveform of an SEE plus timing simulation of a 2-input NAND gate. It displays the generation of three different SEEs, each one with a different value of pulse width and thus with a different charge value. The first SEE has the same width as the critical charge, and an equivalent pulse is generated at the output of the cell. The second SEE width does not convey enough charge to create an event on the output of the cell. Thus, it is filtered. The third SEE has a pulse width capable to generate an event at the cell output.

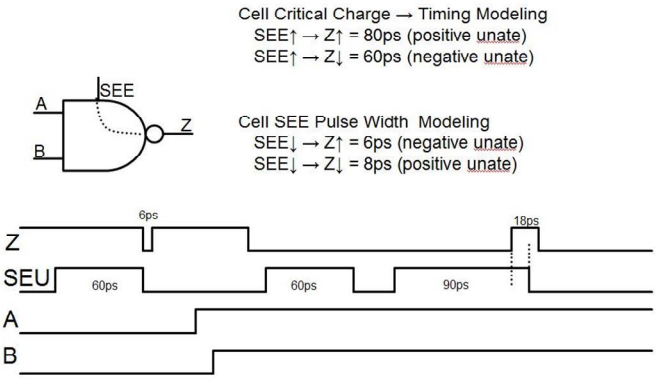


Figure 8 – Waveform of an SEE example simulation for a 2-input NAND.

E. SEE Optimization

The results generated by the SEE cell characterization process can be used in other steps of the design flow besides the fault simulator. One way to use it is applying some rules for swapping the cells in the design, to obtain a more robust system. In this work, this is done by ranking cells with the same function based on the critical charge. Cell swapping is followed by a new step of static timing analysis to guarantee that the new circuit fits the circuit's timing constraints.

V. RESULTS

The results presented in this Section were divided into SEE characterization results and digital flow results. Both are extracted for a 32nm technology under 1V, 27°C and typical transistor models. The exponential source was modeled using parameters extracted from the literature including [7] and [6]. Since no physical radiation effect characterization of the 32nm technology is available in the literature so far, the timing constants of the exponential sources were tailored to be pessimistic, based on previous bulk technology ($\tau\alpha$ = 100ps and $\tau\beta$ = 1ns).

A. SEE Characterization Results

Figure 9 shows the relationship between normalized critical charge and the drive strength for cells in a 32nm standard cell library, when driving a 5fF output load. The minimum value is 0.0105 for a 4-input multiplexer with drive strength of 0.0940, while the maximum value is for an inverter with drive strength of 1.57pF. The critical charge is correlated with the drive strength and consequently with transistor sizing, but is highly dependent of the employed transistor configurations. For example, to drive high load capacitance, it is normal to apply transistor folding or associating a chain of buffers to the cell output. The second strategy is more susceptible to SEE, since the buffer chain has the capacity to regenerate the signal. In this way, the designed SEE cell cannot be based just on cell sizing.

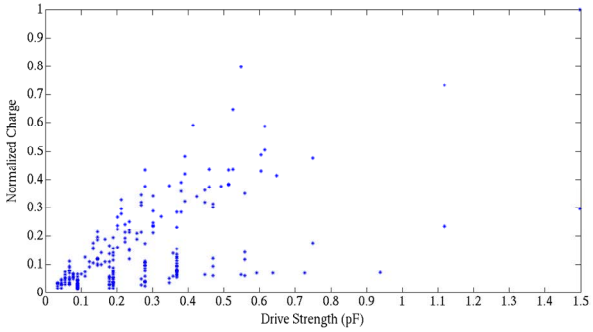


Figure 9 – Relationship between the drive strength and the normalized critical charge. Normaliztion was performed with the critical charge value of the most robust cell in library.

Figure 10 shows the standard cell library normalized critical charge distribution. Based on these results, it is possible anyway to build subsets of the cell library to improve circuit robustness.

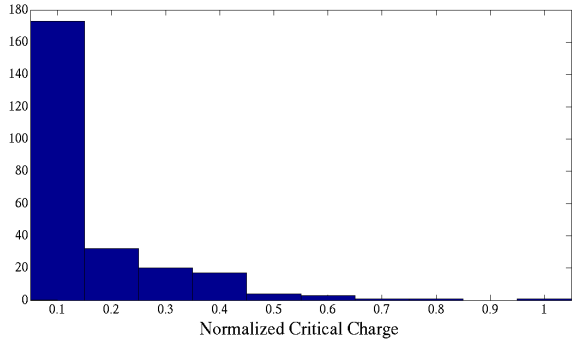


Figure 10 – Histogram of normalized critical charge in a 32nm standard cell library. Normaliztion was performed with the critical charge value of the most robust cell in library.

B. Testcase Results

To validate the proposed flow, an RSA cryptographic core was implemented and validated. Cryptographic circuits are good candidates to hardening techniques, since robustness is one of the main requisites of such systems. Table 1 shows the

area results obtained after the synthesis of the RSA core. Tests were performed with the circuit operating at 200MHz and with successive encryption and decryption operations.

TABLE 1 – AREA CHARACTERISTICS OF THE RSA CRYPTOGRAPHIC CORE.

Cell Type	Instances	Area (μm^2)	Area (%)
Sequential	1439	8219.5	14
Inverter	3742	1552.3	2.6
Buffer	462	438.8	0.7
Logic	48196	48447.2	82.6
Total	53839	58658	100

Figure 11 depicts the relationship between the normalized charge applied to the RSA core and the interval between failures. In the Figure, two different implementations of the RSA Core are presented. The first one is the implementation obtained from the commercial synthesis tool. The second implementation was obtained through cell swapping. For small charges both implementations have the same susceptibility. However, for higher levels of applied charge the SEE hardened implementation presents as much as 100% larger time between failures as the RSA core generated automatically by the synthesis tool. The oscillation on the RSA core curve is due to the correlation of the SEE generation and the clock signal in some cases. SEE insertion rate is constant and equal to 5µs.

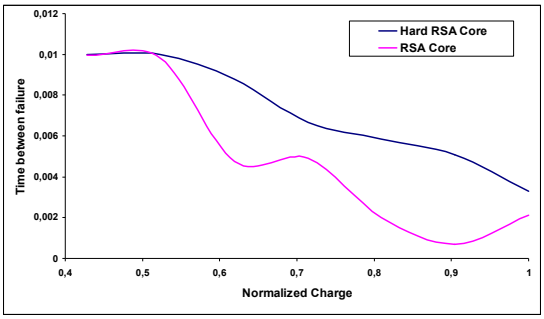


Figure 11 – Relationship between applied charge and the time between failures (in seconds) for the RSA core and a SEE Hardened RSA core.

Figure 12 shows the relation between the SEE injection rate through the fault simulator and the time between failures. Oscillations are caused by clock and SEE generation relationship in the analyzed period. Injected Charge Results are normalized to the Maximum Charge injected. The values of rate injection and critical charge for the system are the input constraints for the SEE design flow. The triplet (failure in time, SEE rate, SEE Charge) is used as a constraint of the presented SEE design flow. For example, the triplet (7ms minimum failure rate, at 5µs SEE interval, with an injected charge of 70% of the maximum injected charge) is an example constraint used for the RSA cryptographic core. For this example, Figure 11 shows that the new hardened version of the RSA is capable to satisfy the constraints.

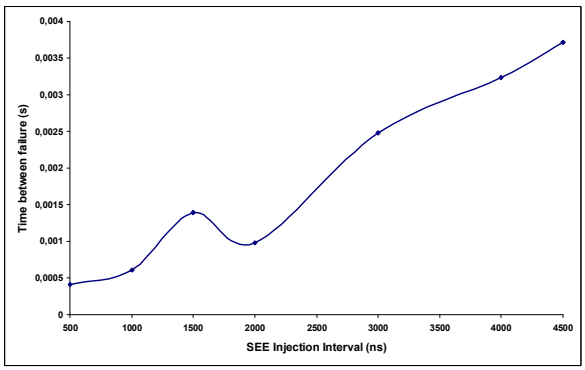


Figure 12 – Relationship between SEE injection and the time between failures for the RSA core.

VI. CONCLUSIONS

In order to increase the robustness of digital circuits to radiation effects it is mandatory to have available tools capable of providing accurate estimations of SEE effects in a fast way. The flow presented in this paper is well aligned with this trend. Results presented in this paper show that the flow can be applied for real SoCs and IPs. The cell characterization process is an important step in the flow to obtain accuracy in the results at the end of the flow. The evaluation of the results accuracy is an ongoing work.

REFERENCES

[1] International Technology Roadmap for Semiconductors, http://www.itrs.net/, 2009.

[2] R. Baumann, "Radiation induced soft errors in advanced semiconductor technologies," IEEE Transactions on Device and Materials Reliability, 5(3), pp.305-316, September 2005.

[3] H. Asadi, M. Tahoori, "Soft error modeling and remediation techniques in ASIC designs," Microeletronics Journal, 41, pp. 506-522, July 2010.

[4] K. Mohanram, "Closed-form simulation and robustness models for SEU-tolerant design," In: *Proc. VTS*, pp. 327-333, 2005.

[5] R. Rao, K. Chopra, D. Blaauw, D. Sylvester, "Computing the soft error rate of a combinational logic circuit using parameterized descriptors," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 26(3), pp. 468-479, March 2007.

[6] M. Gaillardin, P. Paillet, V. Ferlet-Cravois, J. Baggio, D. McMorrow, O. Faynot, C. Jahan, L, Tosti, S. Cristoloveanu, "Transient radiation response of single and multiple gate FD SOI transistors," IEEE Transactions on Nuclear Science, 54(6), pp. 2355-2362, December 2007.

[7] V. Ferlet-Cavrois, P. Paillet, D. McMorrow, J. Melinger, A. Campbell, M. Gaillardin, O. Faynot, O. Thomas, "Analysis of the transient response of high performance 50-nm partially depleted SOI transistors using a laser probing technique," IEEE Transactions on Nuclear Science, 53(4), pp. 1825-1833, August 2006.

[8] M. Omaña, D. Rossi, C. Metra, "Model for transient fault susceptibility of combinational circuits," Journal of Eletronic Testing: Theory and Applications, 20(5), 501-509, October 2004.

[9] F. Wang, Y. Xie, "Soft error rate analysis for combinational logic using an accurate eletrical masking model," IEEE Transactions on Dependable and Secure Computing, 8(1), pp. 137-146, February 2011.

[10] B. Zhang, W. Wang, M. Orshansky, "FASER: Fast analysis of soft error susceptiblity for cell-based designs," In: *Proc. ISQED*, pp. 755-760, 2006.

[11] Open Source Liberty, http://www.opensourceliberty.org/, 2011.