AFTU, an Analog Single Event Effects Automatic Analysis Tool

F. Márquez [a], L. Sanz [a], F.R. Palomo [a], F. Munoz [a] and M.A. Aguirre [a]

[a] Grupo de Ingeniería Electrónica (GIE), Dpto. de Ingeniería Electrónica,
Escuela Superior de Ingenieros, University of Seville, 41092, Seville, Spain

fernando.marquez@gie.us.es, fmunoz@us.es

Abstract

The Analog FTU Hardware Debugging System (AFTU) is an End-User Development (EDU) software tool for the analysis of Single Event Effects on microelectronic designs. AFTU takes user inputs as the design netlist, VLSI technology type, injection points and analysis heuristics classes to generate an expert software with two main functions: management of a simulation set through Spectre and a heuristic based inference engine to classify and analyze the simulation results.

The actual AFTU 1.0 simulates single event effects through current injection models. The simulation sets of results are organized by the inference engine and selected under heuristic based inferences, in order to present a Single Event Effects (SEE) vulnerability assessment to the microelectronics designer.

I. Introduction

With the scaling of CMOS technologies, microelectronic devices have increased their error sensitivity, demanding more attention for the design of radiation-hardened systems. In many applications, such as nuclear or spatial electronic systems, circuits can be exposed to high energy particles which can lead to Single Event Effects (SEE). In modern technologies, a well-known threat such as Single Event Transient (SET) errors generated from heavy-ion strikes is becoming even more influential nowadays [1]. Tools like TCAD [2] can extract exhaustive and accurate information of the effects on the proper design layout. However, dealing with the analysis of complex analog circuits is a challenging task, especially when the number of transistors increases leading to great computational costs.

In this sense, a systematic method of analysis for radiation tests in analog circuits by means of Spectre-based simulation tools was proposed [3]. Following previous experiences on digital technologies -like the European Space Agency (ESA) project FT-UNSHADES, [4]-, a fault injection simulation tool -AFTU- for SET analysis of analog and mixed-signal circuits is currently being developed by GIE (University of Seville) under ESA activities. This tool automatically modifies the circuit netlist, adding configurable current sources emulating the current injection produced by particle impacts. The ionization model applied in electrical simulation is a current source with double exponential dynamics, as illustrated in the next equation:

$$I_{rad} = \frac{Q_c}{\tau_d - \tau_r}\left(e^{-\frac{\tau}{\tau_d}} - e^{-\frac{\tau}{\tau_r}}\right) \quad (1)$$

where τ_r is the rise time related to the plasma track dynamics, τ_d is the down time related to charge drift and diffusion in the transistor, and Qc is the net charge associated to the transient current through the transistor node. This is a well-known model widely described in literature [5] that can be implemented using a VerilogA model.

As a particular case of study, a D-latch with some combinational logic in a 130 nm CMOS technology of ST Microelectronics has been analyzed using AFTU to determine the critical charge required to generate a Single Event Upset (SEU) at its output.

II. Tool Description

The Analog FTU Hardware Debugging System is a tool to evaluate the SEE sensitivity of analog/mixed signal circuits at transistor level. To perform this task, the tool takes an Spectre netlist from the circuit under test and emulates radiation conditions by means of adding configurable sources. The user can define all the required parameters to perform a test campaign using some configuration files, and the tool will automatically apply the selected heuristics for SET sensitivity analysis and generate an output file with statistical results allowing a vulnerability study of the target circuit.

From a given netlist, extracted from a user transient test-bench, the tool will automatically create an instrumentalized netlist with incorporated SET injection models and an identically functional performance. Using this netlist as a starting point, it generates Ocean-based scripts to inject SETs to the circuits under test and extract the information from their performance. Using Cadence OCEAN Scripts, the ionizing particle impacts can be simulated in Spectre at every selected node and time chosen by the user. The generated scripts allow performing a set of parametric simulations dependent on the intensity of the particle impacts in one (or several) nodes at different times chosen by the user.

The tool allows several global parameters for simulation and analysis to be configured by the user:

- Devices where an impact on any transistor of the design should be emulated. The user is able to consider all possible devices or to focus the campaign in a specific part of the circuit.
- Total amount of charge injected for every selected node where an impact is emulated.
- Times in which the impacts are considered to be emulated. For a same node, different impact times allow the circuit evaluation in all possible working points.

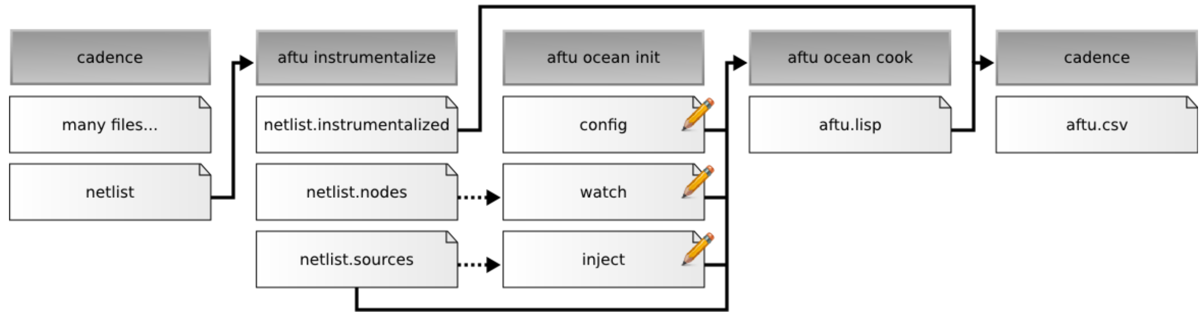

Figure 1: Analog FTU toolchain

- Circuit nodes in which the effects of a possible SET should be analyzed and reported. Differential signals and algebraic expressions can be considered in addition to single-ended outputs.
- Parameters related to the analysis of the signals, such as voltage thresholds, simulation time, simulation step, signals range definitions, etc.
- Information about the technology employed (dependent on the design kit used)

With this information, the response of the circuit under test with injected SETs is compared to a non-irradiated version of the considered outputs, determining the most relevant information to the user (designer of the circuit).

III. AFTU TOOLCHAIN

In Figure 1, the Analog FTU toolchain is presented, comprising a user interface, a full compiler from user input to code generator in SKILL language and an end analysis presentation file. The output code is the expert software, to be interpreted by Cadence OCEAN. The process simulation flow, commanded by the expert software, generates a set of simulations of the microelectronic design under several SEE situations, coded as injection models, in different design elements.

For a target circuit to be analyzed, the tool requires the netlist generated from a Spectre transient simulation as an input. This file can be taken from the original test bench used by the designer to test the circuit functionality. The tool will automatically generate (aftu-instrumentalize) an instrumented version of the netlist with added SET injection models that do not modify the circuit properties but allows to emulate radiation conditions, and two files with all the required information of available transistors where an impact can be emulated (*netlist.sources*) and observable nodes where the effects of injected SETs can be evaluated (*netlist.nodes*).

After this step, aftu-ocean-init generates three files (*config, inject, watch*) that can be used by the user as a template to define all the necessary parameters and criteria to define the test campaign. The information contained in *netlist.nodes* and *netlist.sources* files can be used as a feedback by the user to complete these files. *Config* file contains necessary information for analysis configuration: paths, times, applied heuristics, initial values, etc. *Watch* file allows the user to define all elements in the circuit to be observed during the simulation. *Inject* file allows to define where, when and how much charge can be injected (radiation emulation).

Once these files have been properly filled by the user, aftu-ocean-cook generates a script-based file (*aftu.lisp*) which includes all the paths and data required and implements the heuristics defining the way the simulation has to be performed and results analyzed. The user will take this file and run it in Cadence to emulate radiation conditions over the target circuit and will obtain a results file (*aftu.csv*) with all the statistical processed data, allowing a SET sensitivity evaluation of the circuit under test at transistor level.

IV. CASE STUDY

The target circuit is a D-latch cell with some combinational logic in a 130 nm CMOS technology of ST Microelectronics, as shown in Figure 2.

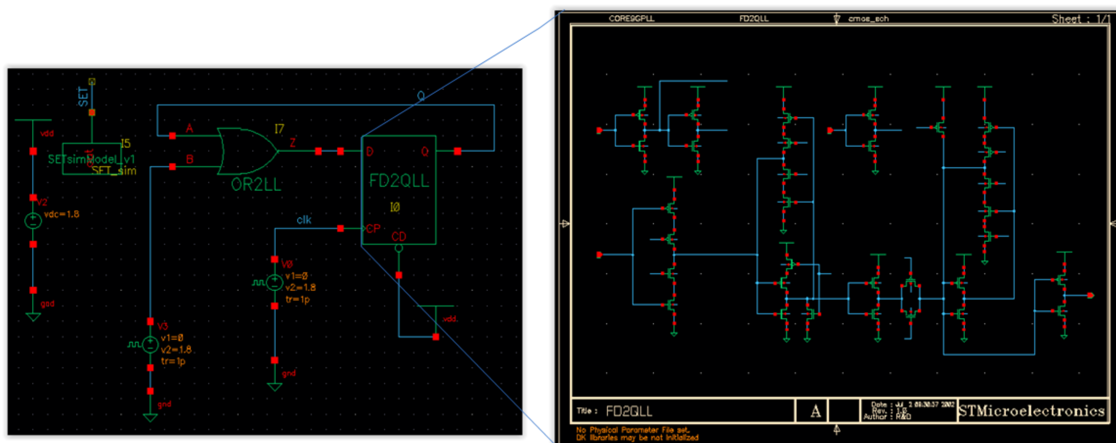

Figure 2: D-latch transient test bench and schematic view of the digital cell

Critical charge will be dependent on several factors such as the injected amount of charge, biasing point of transistors, emulated impact times, transistors affected, etc. Using AFTU, an analysis of different injected charges in every transistor in every possible impact time can be performed, as described in next paragraphs.

For the analysis of this circuit, all the required parameters will be properly configured in the *config* file, setting a 3 ns simulation (three signal periods) time with the appropriate analysis heuristic and including all the necessary paths and parameters for simulation. The applied heuristic is based on defining an error threshold for every observable signal to measure its maximum deviation (referred to the non-irradiated signal) as a consequence of emulated impacts and the recovery time in which its value can be greater than the user-defined threshold (Figure 3).

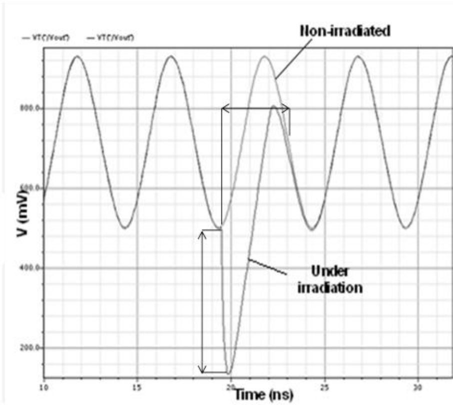

Figure 3: Applied heuristic description

The idea of the proposed analysis is to evaluate the response of the latch during three periods (3 ns): the first one is for signals initialization and settling, the second for injection of emulated SETs and the third for observation of the selected outputs. This test campaign will be defined by means of the *watch* and *inject* files.

On the one hand, we will select the observable signals to watch during the performed SET analysis in the *watch* file:

```
watch Q = /Q :
        threshold = 0.975 ;
```

Figure 4: Format of the *watch* file

where the name of the output (Q) is extracted from the correspondent *netlist.sourcelist* file. The threshold has been set to 0.975, as it is the value for high-low commutation in this latch. In this way, when the signal changes from high to low state (or vice versa) it will be detected and processed by the heuristics applied

On the other, hand, the *inject* file (Figure 5) is defined to select the different values of charge that will be injected in every selected transistor of the target circuit. The idea is to perform an injection for different values of charge in every transistor every 0.1 ns during one signal period. In this way, the SEE sensitivity can be evaluated by determining the required value of charge as a function of the impact time to generate a SEU at the output. With the analysis of these data given by the generated script, a SEU probability can be estimated with a 10% precision (as ten impacts per period are being emulated) for one signal period. As previously stated, information of the available sources for charge injection can be taken from the *netlist.sources* file generated using aftu-instrumentalize.

```
inject I0_MN20:
        Q = .025p, .05p, 0.1p, .2p, .5p, .75p, 1p, 1.5p;
        t = 1.0n : 1.9n : 0.1n;
inject I0_MP19:
        Q = .025p, .05p, 0.1p, .2p, .5p, .75p, 1p, 1.5p;
        t = 1.0n : 1.9n : 0.1n;
inject I0_MN19:
        Q = .025p, .05p, 0.1p, .2p, .5p, .75p, 1p, 1.5p;
        t = 1.0n : 1.9n : 0.1n;
inject I0_MP18:
        Q = .025p, .05p, 0.1p, .2p, .5p, .75p, 1p, 1.5p;
        t = 1.0n : 1.9n : 0.1n;
...
```

Figure 5: Format of the *inject* file

The script *aftu.lisp* is generated and after its execution in Cadence, a *results.csv* file is generated that allows obtaining the necessary information to evaluate the SEU sensitivity of the cell. A representative part of this results file is shown in Figure 6:

Output	ImpactNode	Qinj	Timp	Trec	Vmax
V_Q	I0_MN11	2.5e-14	1e-09	0.0000	0.006827
V_Q	I0_MN11	2.5e-14	1.9e-09	0.0000	0.016568
V_Q	I0_MN11	5e-14	1e-09	0.0000	0.017084
			. . .		
V_Q	I0_MN11	5e-14	1.3e-09	0.0000	0.005371
V_Q	I0_MN11	5e-14	1.4e-09	1.6100	1.806680
V_Q	I0_MN11	5e-14	1.5e-09	1.5000	1.806814
V_Q	I0_MN11	5e-14	1.6e-09	1.4000	1.805740
V_Q	I0_MN11	5e-14	1.7e-09	1.3000	1.802925
V_Q	I0_MN11	5e-14	1.8e-09	0.2400	1.404223
			. . .		

Figure 6: Format of the *results.csv* file

For the case a low level output is expected, it can be observed that there are several values of recovery time (T_{rec}) greater than 1 ns. This means there has been a change at the output which has been captured by the latch, generating a SEU at the output.

Considering the ten impact instants emulated for every transistor and different values of charge, a SEU probability can be estimated for the injection period. Taking the MN11 transistor as an example, for an injected charge of 0.05 pC there are generated SEUs for five impact times (from 1.4 to 1.8ns), leading to a 50% of SEU probability for this value of charge. Extending this evaluation to the rest of transistors of the circuit, it is possible to obtain a sensitivity map of the resulting SEUs under irradiation conditions for the target circuit analyzed using AFTU, as shown in Figure 7 for the considered case study.

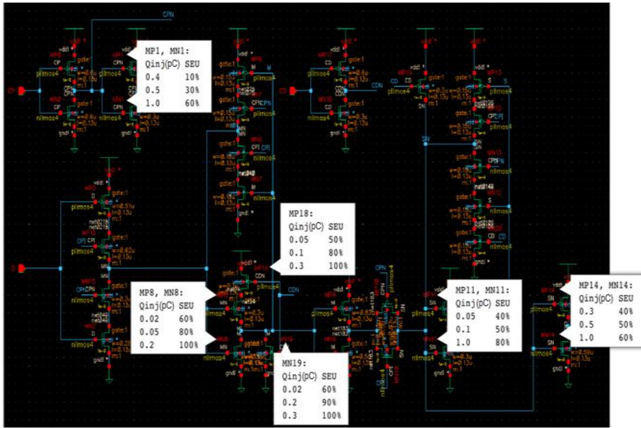

Figure 7: SEE sensitivity map of the target circuit

V. CONCLUSIONS

A tool for automatic analysis of analog/mixed signal circuits affected by radiation has been developed and tested. AFTU allows impact emulation, based in current injection models, in every transistor of a given design by means of configuration files edited by the user. The analysis of every node of the circuit and different signals defined by the designer can be performed applying several heuristics in different available technologies. Thanks to the automated placement and script generation, massive injection campaigns can be performed over target circuits to diagnose their SEE sensitivity following heuristic criteria for error discrimination defined by the user.

VI. ACKNOWLEDGEMENTS

This work was partially supported by the Spanish Ministry of Economy and Competitiveness under the project TEC2010-22095-C03-01 and developed under a collaboration agreement between the European Space Agency and University of Seville.

VII. REFERENCES

[1] P. Jaulent, V. Pouget, D. Lewis and P. Fouillat, “Study of Single-Event Transients in High-Speed Operational Amplifiers”. IEEE Transactions on nuclear science, vol.55, nº 4, Aug. 2008

[2] J.M. Mogollón, F.R. Palomo, M.A. Aguirre, J. Napoles, H. Guzmán and E. Garcia, “TCAD Simulations on CMOS Propagation Induced Pulse Broadening Effect: Dependence Analysis on the Threshold Voltage”, IEEE Transactions on nuclear science, vol.57, nº 4, Aug. 2010

[3] Marquez, F.; Munoz, F.; Palomo, F.R.; Aguirre, M.A. and Ullan, M., "Analysis of Single Event Transient Effects in Analogue Topologies" 4th International Workshop on Analogue and Mixed-Signal Integrated Circuits for Space Applications (AMICSA ‘12). Aug. 2012.

[4] Aguirre, M.A.; Tombs, J.N; Muñoz, F.; Baena, V.; Guzman, H.; Nápoles, J.; Torralba, A.; Fernández-León,A; Tortosa-López, F. and Merodio, D., "Selective protection analysis using a SEU emulator: testing protocol and case study over the LEON2 processor," IEEE Transactions on Nuclear Science, 54(4), pp.951,956, 19-21 Sept. 2007.

[5] G. Messenger, “Collection of Charge on junction nodes from ion tracks”, IEEE Transactions on nuclear science, vol.29, nº 6, Dec. 1982.