A Tool to Evaluate Stuck-Open Faults in CMOS Logic Gates

Alexandra L. Zimpeck, Cristina Meinhardt, Paulo F. Butzen

Center for Computational Science – C3
Universidade Federal do Rio Grande – FURG Rio Grande – RS, Brazil
{alexandra.zimpeck, cristinameinhardt, paulobutzen} @furg.br

***Abstract*— A tool that simulates Stuck-Open Fault (SOF) is presented in this paper. In the nanoscale technology context, the behavior of Stuck-Open Fault is more affected by the high leakage currents. To analyze SOFs is necessary one pair of vectors, because the fault effect depends on previous state. This tool allows the circuit behavior analyses in the presence of SOFs in CMOS logic gates. Through the test vectors used in the simulation it is possible to compute the percentage of detecting faults during the test application. Also, the holding time analyses can be performed.**

***Keywords*— Stuck-Open fault, Leakage currents, CAD tools.**

I. INTRODUCTION

The advent of integrated circuits was only possible due to the technology scaling which allowed the integration of more transistors in a single chip. This higher number of devices in the same area generates an increase in circuit functionality and performance. However, the technology scaling causes several undesired effects, as variability, aging effects, and leakage currents.

Another undesired behavior verified in nanoscaled technologies is a significant increase in the number of possible faults [2]. In this subject, the concepts of fault, failure, error and defect are extremely important for a full comprehension.

In this work, the concepts of faults, failure, errors and defects are considered as defined in [3]. Fault is an incorrect operation of a system, and can be originated from design mistakes, physical defects or external interference. It can be also said that faults are an unexpected condition that can lead the system to achieve abnormal states. A fault can lead to an error. Error is defined as an undesired change in the state of system. The presence of an error can lead to an incorrect response of the system, which is named failure. Defects are considered deviations of the specification. Concluding, a single fault can cause multiple errors in a system, and an single error can cause system malfunction.

This article focus on an open defect type in MOS transistors called Stuck-Open Fault (SOF). Opens in a single transistor gate may compromise noise margin, speed of operation, and quiescent power supply current [4]. SOFs have been extensively explored in 1980s [5][6]. In recent years, this type of fault becomes a relevant defect mechanism due to the interaction to high leakage currents verified in modern circuits.

SOFs are difficult to be tested because the output state depends on the energy load at the output capacitance in the previous state. To verify SOFs is required specifics 2-vector pairs [7]. The high leakage currents in nanoscale technologies influence greatly the well know behavior of SOFs.

To evaluate the behavior of the circuits in the presence of faults is important the use of tools that assist the project in nanometer technologies. Therefore, this work presents a tool that allows the fault SOFs simulation in CMOS logic gates. This tool analyzes the circuit behavior in the presence of SOFs and the interaction with leakage currents, also computing the holding time parameter. Through the test vectors used in the simulation is possible to compute the percentage of detected faults during the test application.

The rest of the paper is organized as follows. Section 2 re-examine, through an example, the SOFs behavior. The importance of leakage currents in SOF analysis in nanometer technologies is discussed in Section 3. Section 4 presents a description of the proposed tool modules and the main development particularities. Some examples of results that can be generated with the toll are presented in Section 5, following by conclusions and futures works presented in Section 6.

II. STUCK-OPEN FAULT

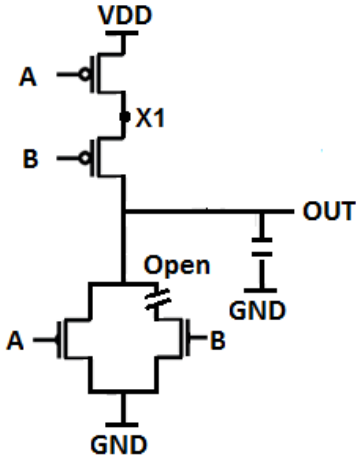
Let consider a digital system where transistors MOS are modeled as keys. In normal operation, a transistor makes the connection between two circuit nodes according to the applied signal at the gate terminal. If a SOF happens in a transistor of this system, it indicates that the connection between two nodes by this transistor will never happen, independently of the signal applied at the gate terminal [8].

To exemplify the SOF effects in a logic gate, Figure 1 shows a NOR2 logic gate and its truth table with the correct output, and also with the wrong output caused by the SOFs. In this example, the SOF occurs in transistor B on the pull-down network. The truth table, with the wrong outputs, was made considering previous state as high.

Looking at each state of the truth table, we get the following behavior: For the AB=00 vector both the PMOS transistors are turned on, leading to output the correct logic value. The second vector AB = 01, the good transistor in the NMOS network is off and the faulty transistor should be turned on, but cannot supply current to output because of the SOF. Therewith, both pull-down network paths are blocked,

making the output signal floats in a high impedance state, maintaining the voltage of the previous state. The value of the previous state is stored in the load capacitance.

For the AB = 10 state, the good NMOS transistor is turned on, taking to output the correct logic value. The same results occurs to AB = 11 vector.



A	B	OUT	OUT*
0	0	1	1
0	1	0	1
1	0	0	0
1	1	0	0

Fig. 1 - Stuck-Open fault in a logic gate NOR2 with good (OUT) and bad (OUT*) truth table response

Figure 2 shows an example of the structure used for analysis of logic gates in presence of faults. In this experiment, a NOR2 logic gate is connected to an inverter and the signal is observed in the output of the gate and after the inverter output. This structure highlights the influence of SOF in the next logical stage. The expected result, i.e., the result obtained without SOF in the logical gate, for the out signal (out) and for the Y signal (out_inv) is shows in Figure 3. For this example, this structure is evaluated considering the pair of vectors AB = 00 and AB = 01 in predictive technology of 45nm [9].

One way to simulate the injection Stuck-Open faults in the pull-up network consists in keep the gate terminal with value 1, let always open the PMOS transistor. When the same strategy is applied to pull-down network, the gate terminal is maintains with value 0, i.e. forcing the transistors NMOS to stay open.

Figure 4 shows the effects of injecting one single stuck-open fault in NOR2 gate at the B transistor of pull-down network. The output signal falls slowly because the output is in high impedance. The next logical level only will receive the expected input after a delay due to the stuck-open fault.

III. STUCK-OPEN FAULTS IN NANOMETER TECHNOLOGIES

Stuck-Open fault behavior must be re-examined in nanometer technologies because nanotransistors present a significant leakage current. This behavior is critical during the SOF high-impedance state [10]. Related works highlight that two dominant leakage components, the gate tunneling leakage and S-D subthreshold leakage, are the main reasons to investigate the SOF behavior at nanotechnologies [11].

For the pair of vectors AB = 00 and AB = 01, remember that the output signal is floating in a high impedance state. The output signal maintains the voltage of the previous state because of the capacitance. In nanotechnologies, this output signal characteristic suffers with the high leakage currents. The output is held in the last state for a considerable short time in relation to the old technologies.

To illustrate this behavior and highlight the influence of leakage current in the SOF detection, the same experiment is done with 45nm and 16nm technologies. Figure 5 illustrate the behavior of the circuit in presence of fault considering predictive technologies of 16nm (out16) and 45nm (out45) [9].

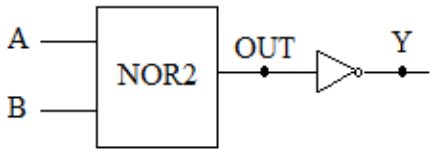


Fig. 2 – NOR2 logic gate with inverter

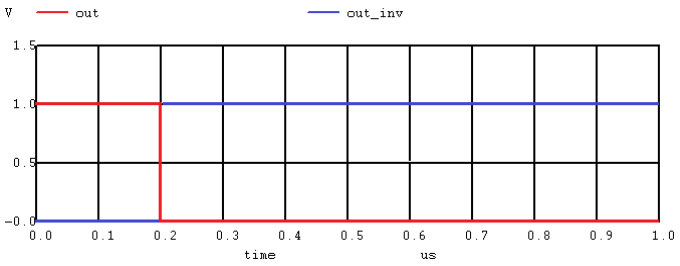


Fig. 3 – Expected result with test application of technology of 45nm

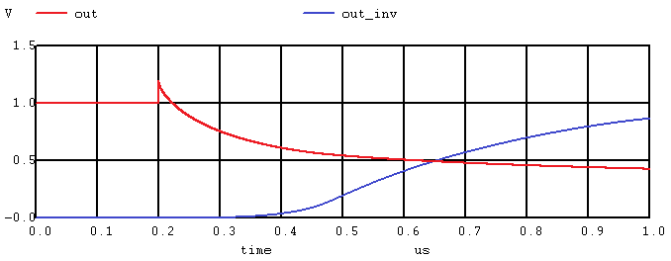


Fig. 4 – Obtained result with SOF in technologies of 45nm

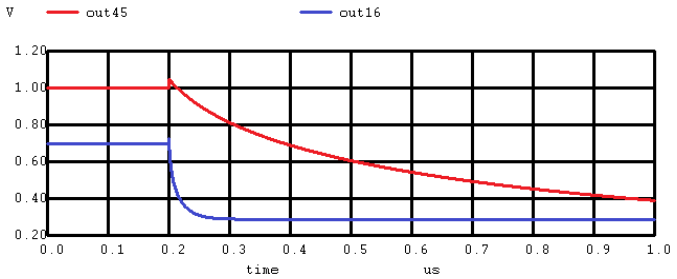


Fig. 5 - Obtained result with SOF in technologies of 16nm and 45nm

One important term to measure the SOFs effects in nanotechnologies is the holding time. The holding time is a measure related to the delay induced by the SOF [8], i.e., the time for node-OUT discharge from VDD to VDD - |Vtp|, where |Vtp| is the PMOS transistor threshold voltage. If we had a charging, the charging time for node-OUT would be from GND to |Vtn|, where |Vtn| is the threshold voltage of NMOS transistor.

In this work, the threshold voltage is considered the value of 0.3V for both technologies. The holding time can also be interpreted as the time that voltage can be interpreted as a logic low state in the next stages.

Considering the previous analyses and observing Figure 5, the holding time for technology of 45 nm is approximately 200ns. For the 16nm technology, the output signal was partially discharged in a short time, so, the holding time is approximately 30ns.

It is important determine the holding time, because, depending on the operation frequency of the circuit, some outputs can be at the correct electrical level or not. Table 1 shows the number of faults obtained from analysis in high frequency (1GHz), medium frequency (10MHz) and low frequency (100kHz) in logic gate NOR2 considering the exhaustive test , i.e., in this example is generated 12 pairs of test vectors. These results were obtained with the tool developed and highlight the influence of leakage currents in the effects of SOFs in nanotechnologies.

Table 1
Number of Faults for 16nm and 45nm at Low, Medium and High Frequency

Technology	Low Frequency	Medium Frequency	High Frequency
16nm	0	0	1
45nm	0	1	1

IV. TOOL DEVELOPMENT

The developing tool was made in a modular method, divided in six main modules, according to the block diagram showed in Figure 6.

The tool was developing in JAVA programming language and the circuits were simulated in the electric simulator NGSpice [12]. The language programming JAVA was chosen because it has portability and platform independence, allowing integration of the other tools/software. The program has six classes and one interface that implement the types of faults. It has 1494 lines of code and each simulation takes about one minute. Each part of the program will have its specific function:

a. User interface: Is the means of communication with the program user. Figure 7 shows the interface of the tool in Portuguese. The input data fields are displaced in the left side of the window. The input data fields available in this tool are the number of inputs of the logic gate evaluated, the technology, the voltage value adopted in the specified technology, the fault type that will be simulated, and the circuit description in SPICE language.

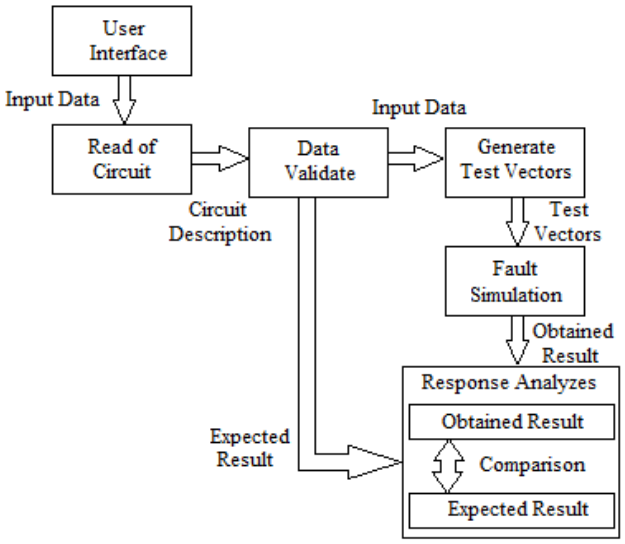


Fig. 6 - Block Diagram of the development tool

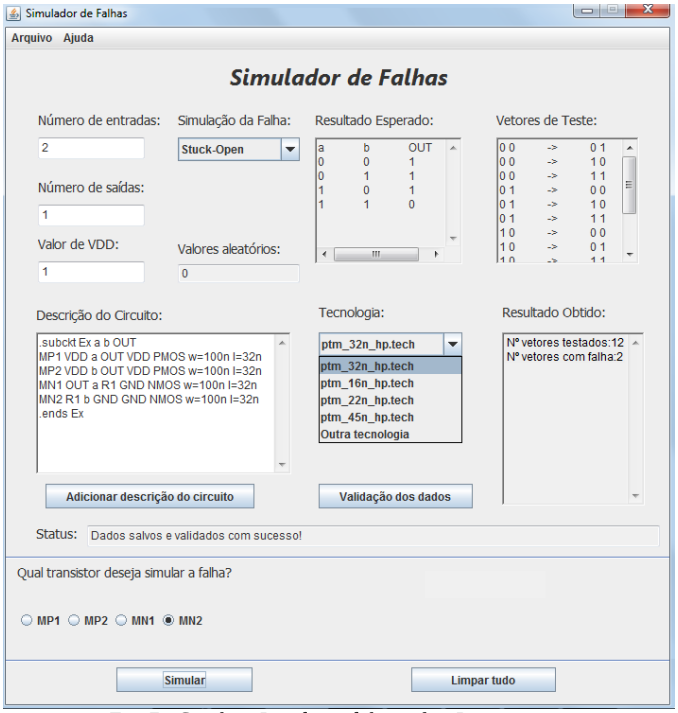


Fig. 7 - Graphics Interface of the tool in Portuguese

At the right, the output values are showed. This tool calculates the correct truth table for the logic gate description and displays it. Moreover, the interface displays the fault injection vectors generated and summarizes the results.

The user can select the transistor to inject the fault, clicking in the check-box generated after reading the circuit description.

b. Read of circuit: Consists in knowing the circuit in which the fault will be simulated. The circuit description format follow the SPICE format to describe subcircuits, i.e., the circuit is composite by a name, by their input and output

terminals and the electronic components (transistors, capacitors, etc) presents in circuit. In figure 8 is shown an example.

```
.subckt NOR2 a b OUT
MP1  X1   a  VDD  VDD PMOS w=100n l=45n
MP2  OUT  b  X1   VDD PMOS w=100n l=45n
MN1  OUT  a  GND  GND NMOS w=100n l=45n
MN2  OUT  b  GND  GND NMOS w=100n l=45n
.ends NOR2
```

Fig. 8 - Example of circuit description accepted by the program

c. Data validate: It makes verification of data supplied by the user. Case these data are correct, it is automatically generated the correct truth table. After the Data Validation, also appears on the tool Interface a field for the user chooses which of transistors present in circuit will be injecting the fault. Case one data is incorrect, is requested the user to enter new data until they pass in the validation step.

d. Generate test vectors: Always generated the test vectors in pairs, since SOFs analyses depend on the previous state because the output is floating in a high impedance state. For the circuit up to three entries, the generation of pairs is exhaustedly, i.e. are generated all possible combinations of vectors. For a circuit over three inputs, the generation of pairs is random, where the user that determines the amount of random vectors that must be generated.

e. Fault simulation: This module simulates the circuit in the presence of a stuck-open fault in the transistor chosen for the user. The circuits will be transparently simulated in NGSpice electric simulator. This fact allows to the user abstract the contact with the electrical simulator.

f. Response Analyzes: This module receives two inputs: the expected result and the obtained result. The expected result is the objective, i.e. the result logically correct for the function. The obtained result is what you can get with the simulation considering the fault injection. This module makes the comparison between these two results and, from this module calculates the percentage of faults during test application (fault coverage).

The tool developed is useful to evaluate the robustness of CMOS logic gates to SOFs. As results, the tool shows the number of pairs of vectors tested, differentiating which causes fault propagation and generated an erroneous result in output. Also, the tool gives the number of pairs of vectors that diverged in the previous state and the next state.

V. CONCLUSIONS AND FUTURE WORKS

This work showed the steps of development for a Stuck-Open faults simulator in CMOS logic circuits developed to allow evaluating the behavior of some circuits in the presence of SOFs.

With the aid of a tool development, it is possible to determine the coverage fault of some circuits and evaluating the impact of fault tolerance techniques of type Stuck-Open.

Next step in the development of this tool is include analyses of other kinds of faults, like transient faults and stuck-on faults. Then also be made optimization of the code.

A tool can be also used to assist in the evaluation the impacts of fault tolerance techniques applied to logic gates when submitted to fault Stuck-Open, Stuck-on and transients.

REFERENCES

[1] C. S. Nunes, *Avaliação de Famílias Lógicas para Circuitos com Baixo Consumo de Potência*, Monograf, C3, FURG, Rio Grande, 2011.

[2] L. Anghel and M. Nicolaidis, *Defects Tolerant Logic Gates for Unreliable Future Nanotechnologies*, Lecture Notes in Computer Science, v. 4507, Springer Berlin / Heidelberg, 2007, pp. 422-429.

[3] D. T. Franco, *Fiabilité Du Signal des Circuits Logiques Combinatoires sous Fautes Simultanées Multiples*, Doctoral Thesis, I'École Nationale Supérieure des Télécommunications, França, 2008.

[4] W. W. Needham, C. Prunty and E. H. Yeoh, *High volume microprocessor test escapes, an analysis of defect our test are missing*, in Proc. 1998 Int. Test Conf., 1998, pp 25-34.

[5] R. Wadsack, *Fault modeling and logic simulation of CMOS and MOS integrated circuits*, Bell Syst. Tech., pp. 1449-1488, May-Jun, 1978.

[6] J. Soden, K. Treece, M. Taylor and C. Hawkins, *CMOS IC stuck-open fault electrical effects and design considerations*, In Int. Test Conf., Aug. 1989, pp. 423-430.

[7] R. Gomez, V. Champac, C. Hawkins and J. Segura, *A Modern Look at the CMOS Stuck-Open Fault*, IEEE, 2009.

[8] G. V. Pereira, *Teste da rede de Interconexões de Field Programmable Analog Arrays*, Master Thesis, PPGC, UFRGS, Porto Alegre, 2005.

[9] W. Zhao, Y. Cao, "New generation of Predictive Technology Model for sub-45nm early design exploration," IEEE Transactions on Electron Devices, vol. 53, no. 11, pp. 2816-2823, November 2006.

[10] A. Keshavarzi, K. Roy and C. Hawkins, *Intrisic leakage in low power deep submicron CMOS IC's*, International Test Conference, pp. 146-155, 1997.

[11] Roy, K.; Mukhopadhyay, S.; Mahmoodi-Meimand, H., Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits, *Proceedings of the IEEE* , vol.91, no.2, pp.305,327, Feb 2003

[12] NGspice. [Online]. Available: http://ngspice.sourceforge.net/.