Parametric Fault Simulation and Test Vector Generation

Khaled Saab, Naim Ben-Hamida, and Bozena Kaminska
Fluence Technology, 8700 SW Creekside Place, Beaverton, OR, 97008

Abstract

Process variation has forever been the major fail cause of analog circuit where small deviations in component values cause large deviations in the measured output parameters. This paper presents a new approach for parametric fault simulation and test vector generation. The proposed approach utilizes the process information and the sensitivity of the circuit principal components in order to generate statistical models of the fault-free and the faulty circuit. The obtained information is then used as a measurement to quantify the testability of the circuit. This approach extended by hard fault testing has been implemented as automated tool set for IC testing called FaultMaxx and TestMaxx.

1. Introduction

Analog MOS-integrated circuit (IC) design has received great attention in the last few years, mainly due to the trend toward mixed analog-digital chips, to low voltage operation and even to artificial neural network. However, due to the technological tolerances, model parameters are subject to randomness. As a consequence, the circuit behavior differ for different runs, different slices of the same run, and for different dice of the same silicon slice [1].

Process variation has forever been the major fail cause of analog circuit. But with the advancement in the fabrication process and the reduction of transistor sizes, process variation is increasing the fail rate of digital circuits as well. These variations are hard to detect since they do not cause failure at all conditions and are mostly affecting the long-term reliability of the circuit.

Detecting these variations and generating test vectors for them, will be the subject of this paper where a general methodology for determining the statistical behavior of the nominal circuit and the faulty circuit is used.

2. Overview

In the early stages, simulations of the integrated circuits were accomplished only with nominal parametric conditions. Soon after, variations in the fabrication process were found to be the primary cause of parametric yield loss. Number of approaches for the characterization of the fluctuation in an IC process were used including the extraction of the worst-worst case conditions of the process. This method requires that each parameter is set to a maximum or minimum (whichever is more degrading the performance) allowable value without regard to the particular parameter correlation. In addition to being quite tedious and time consuming, the analysis resulted in overly designed products.

Other methods took advantage of the devices correlation where the circuit “slow” and “fast” corners of the MOS transistors are simulated. This method gives a much more realistic description and requires only six corner analysis.

However, analog and mixed circuit analysis require considering parametric variations on resistance, capacitance, and even transmission lines. This new requirement increases the number of corners (since we need to consider all combinations of those elements to find the worst case) and thus more Monte Carlo simulations are needed to capture the circuit behavior.

In order to reduce the required number of Monte Carlo simulations, some ([2] and [3]) suggested evaluating circuit sensitivity and using the obtained information as guide lines to deterministically identifying the circuit corners. Once the circuit corners have been identified, the appropriate reduced number of Monte Carlo simulations for corner analysis can be executed.

But still, what is an analog fault? And what is an analog fault model?

In an attempt to answer those questions, structural testing has been introduced to the analog domain. In [4] and [5] it has been suggested that the circuit could be tested by testing all of its device parameters. If all device parameters are

within their tolerance ranges the circuit is classified as fault-free. As a consequence, the analog fault was defined as a deviation of the device parameter outside of its predefined range of operation.

Manoj Sachdev [7] proposed a realistic defect oriented testability methodology that creates a signature of the fault free circuit in a multi-dimensional space. If the faulty circuit signature exhibits a response outside this space at least by one of the test stimuli, then it is recognized as faulty. However, the concepts of pass and fail in analog circuit is not clear-cut, thus no absolute signature could be generated for comparison.

In [8] a fault simulator for linear analog circuits has been presented. Fault simulation is achieved by abstracting the analog circuit at the behavioral level, and then transforming it from the continuous Laplace domain into the discrete Z-domain.

In [4] a first approach for test vectors generation based on structural testing and on the device parameter sensitivities has been proposed. Indeed, sensitivity has been computed for each device in the circuit. The stimulus with highest sensitivity has been taken as the test vector.

In [2] and [6] similar approach was proposed, but instead of relying only on the individual devices sensitivities information, device variations and output tolerances where taken into consideration as well.

In this paper we present an efficient and realistic approach for parametric faults simulation and test vector generation. Indeed, the process information and the sensitivity to the circuit principal components are used in order to generate statistical models of the circuit: one for the fault-free circuit and one for each fault in the fault list. The statistical models of the fault free circuit and the faulty circuits are used to estimate the probability of accepting a faulty circuit and the probability of rejecting a good circuit. The obtained information is then used as a measurement to quantify the circuit testability.

3. Approach

Due to the complexity of analog signal, efficient analog and mixed circuit testing could only be approached by statistical analysis of the circuit. From the device parameter distribution data and the sensitivity computation results it is possible to compute not only the fault free output parameter distribution but also the faulty output parameter distribution. The proposed approach is divided into eight steps (*Figure 1*):

1. Compute output parameter sensitivity with respect to all device model parameters in a circuit.
2. Obtain the device statistical data for each device model parameter in a circuit.
3. Compute the circuit fault free output parameter distribution when all device model parameters vary within the tolerance margin defined by the device statistical data. Mapping the device variation to performance variation performs this step.
4. Construct the fault list. Fault list consists of all or a subset of the device components and model parameters.

For each fault in the fault list:

5. Perform fault injection by replacing the fault-free device model parameters by its faulty model.
6. Compute the circuit faulty output parameter distribution when all device parameters (except the faulty device parameters) vary within the tolerance margin defined by the device statistical data. This step performs the mapping of the device variation and of the fault effect to performance variation.
7. From the fault free and faulty circuit distributions quantify the degree of testability of the fault. In this step false accept and false reject concept is used as a measure of the circuit testability.
8. Based on the above information, for each fault, the test vector is obtained by choosing the stimulus that minimizes the false accept and/or the false reject.

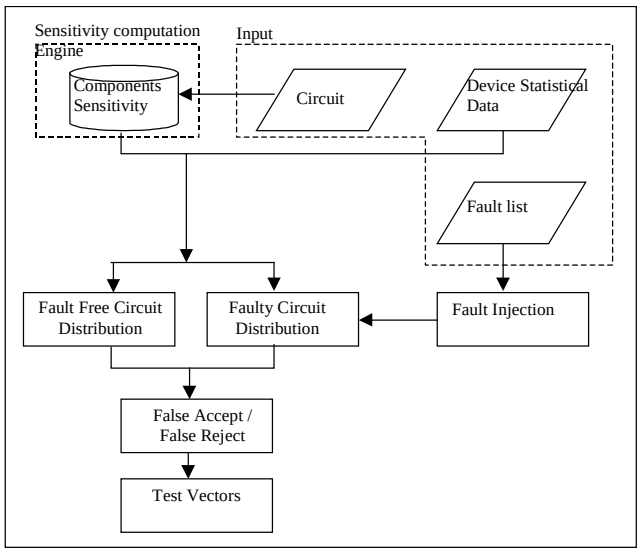

Figure 1: Soft Fault Testing Flow Chart

This paper is organized as fallow: first, the device statistical data will be described. Second we will present how a device statistical data is used to predict the output parameter variation. Then in paragraphs 5, 6 and 7, the devices statistical data and the fault list are used to predict the fault-free and the faulty circuit distribution respectively. Finally, from the fault-free and the faulty output distributions, the false accept and false reject are derived as a measuring criteria in order to obtain the circuit test vectors. Practical examples and results are included in the two last sections.

4. Device statistical data

It is known that the statistical variations of device model parameters are not completely independent. An extremely high degree of correlation is found between devices within a given circuit die. Devices within an individual circuit have experienced the same processing environment, same mask alignment, and have similar electrical performance. Much larger variations are observed between devices on different areas of the slice, or between devices on other slices, or different lots. These inter-die fluctuations result in what we call the process variations. The variations within an individual die (intra-die) are called mismatch variations.

In several cases, the variations within the individual die are significantly less than the die-to-die variations and the mismatch could be neglected. However, in state of the art IC design, die-to-die process variations are minimized and only the device mismatch is affecting the circuit performances thus becoming the primary cause of yield loss.

Thus, properly defining and using the device statistical data is critical since it is the point of convergence between foundry and circuit characterization. The parameter statistics stored in the archive are made of those commonly measured during process (means, standard deviation, and correlation coefficients).

In [9] it has been concluded that for MOSFET characterization only four statistical variables need to be used. These four variables are the geometrical parameters (length, width, and the oxide thickness) and the electrical parameter (flat band voltage). Additional device model parameters, required to represent the device I-V and C-V characteristics, can be described by simple functions of these four independent variables. The sensitivity to other parameters such as change in the doping profile are calculated to be at least an order of magnitude smaller. It is reasonable to expect that these four physical quantities are the principal factors that dominate the statistical distribution of both current and capacitance.

While these studies successfully capture the effect of devices variations, the resultant equations are not compatible with present circuit simulators. In the next paragraphs we are going to present a generic method (simulator independent) that accurately map the device statistical data to output parameter statistical data.

5. Fault free circuit

Because of the uncontrollable variations in device model parameter values inherent to the manufacturing process, the circuit output usually deviates from the nominal response. As long as these deviations are within their specifications, the circuit is declared as fault-free for all functional testing purposes. For structural testing, this definition is extended to the circuit devices where a fault-free circuit is not only defined by its output specifications but by its device specifications as well. The device fault-free specifications are defined to guaranty a fault-free behavior of the circuit under all operation conditions (process variation, temperature variation, radiation…).

Starting from the nominal circuit and using the device statistical data and a standard Monte Carlo approach, circuit output mean and standard deviations can be simulated. However, computational requirements could easily become prohibitively expansive as the circuit size grows and the number of the variable parameters increases.

Our objective is to find realistic, accurate, and efficient solution that allows obtaining the statistical information on the fault-free circuit outputs.

5.1. Fault free circuit distribution

We are interested in finding a general case where we want to describe the effect of a large number of random devices on a given output (*Figure 2*). A piece wise linear estimation of the fault-free circuit output is used. Let *out* be the output parameter of interest such that *out* is a function of x_i ($i = 1...N$). The Taylor series generated by $f(x_0, x_1 \cdots, x_N)$ up to the first order is computed:

$$out = f(x_0, x_1 \cdots, x_N) = out_0 + \sum_{i=1}^{N} S_{x_i}^{out} \Delta x_i \quad \textbf{(1)}$$

Where *out* is the estimated value of the output parameter due to device variation, out_0 is the nominal output when all devices parameters are at their nominal value, Δx_i the variation of the circuit device parameter x_i, and N is the number of variable parameters in the circuit.

From equation (1), the desired output distribution function of the fault-free circuit is obtained by

$$\mu_{out} = \mu_{out_0} + \sum_{i=1}^{N} S_{x_i}^{out} \mu_{\Delta x_i} = \mu_{out_0} \quad \textbf{(2)}$$

$$\sigma_{out} = \sum_{i=1}^{N} (S_{x_i}^{out})^2 \sigma^2_{x_i} + \sum_{i=1}^{N} \sum_{j=1, i \neq j}^{N} S_{x_i}^{out} S_{x_j}^{out} \sigma_{x_i x_j} \quad \textbf{(3)}$$

Where σ_{x_i} is the standard deviation of the device parameter x_i, and $\sigma_{x_i x_j}$ is the covariance term of [x_i , x_j].

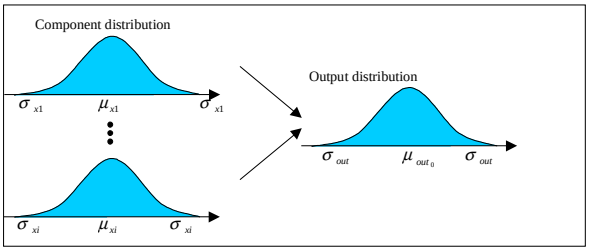

Figure 2: Fault-free circuit distribution

6. Fault list

The fault list is composed of all or a subset of the circuit components and model parameters. Typically, resistance values, capacitance values, the MOS geometric parameters (width, length, and oxide thickness), and the MOS electrical parameter (flat band voltage) are considered. Additional parameters that could affect the circuit behavior (temperature for example) could be considered as well.

7. Faulty circuit

The definition of the faulty circuit depends largely on the definition of a fault.

Presently, in most application areas, fault-free analog circuit and faulty analog circuit classification is based on functional specifications. A faulty analog circuit is defined as a circuit where the sum of all variations under any condition, causes the output to deviate outside of its predefined range. But, since there is no fault model and no fault list, this classification scheme do not give any indication on the fault coverage. Not only that but with no fault list there is no way to know if we are over testing or under testing the circuit.

To overcome this limitation structural testing is used. Structural testing in the analog domain gained more attention in the early nineties [4]. It has been proposed that the circuit could be tested by testing all of its device parameters. If all device parameters are within their tolerance range the circuit is classified as fault-free. While the same approach has widely been accepted for testing digital circuits its impact on analog circuit has been rather controversial until now.

One of the problems that kept this solution at bay is that for each fault there is need to perform fault injection followed by a full fault simulation. In the case of *M* faults and *K* iterations per fault (*K* is the number of Monte Carlo simulations) if standard Monte Carlo approaches are used the total number of Monte Carlo simulations needed is O(*K.M*) which is not realistic for large circuits with large number of faults.

To correct this problem and make the structural testing affordable, here again, for any fault, our objective is to find realistic, accurate, and efficient solution that allows to obtain statistical information on the circuit outputs due to that fault.

Two typical steps are needed. For each fault in a fault list perform:

- Fault injection
- Fast fault simulation

In our case fault injection consists on replacing the fault-free device parameter statistical data by its faulty model. Fault simulation consists on computing the fault output PDF function using a statistical approach. First, fault injection process will be described followed by the computation of the faulty circuit output distribution function.

7.1. Fault injection

First we need to define a fault model. In our case, two fault models could be used:

- The standard single fault model and
- The group fault model.

For the single fault model case only one faulty component is considered during fault simulation. Let $X_p \sim N(\mu_{X_p}, \sigma^2{}_{X_p})$ be the target fault. As long as X_p value is inside the tolerance range $\mu_{X_p} - 3\sigma_{X_p} < X_p < \mu_{X_p} + 3\sigma_{X_p}$ the device parameter is considered as fault-free and thus the circuit is considered as fault free. Setting X_p value to be outside the tolerance range performs fault injection. In other term, changing the X_p distribution from $X_p \sim N(\mu_{X_p}, \sigma^2{}_{X_p})$ to $X_p \sim N(\mu_{X_p} \pm 3\sigma_{X_p}, 0)$ performs single fault injection (see Figure 3).

However, in most cases single fault model is inadequate to represent the silicon faults. Indeed, some of the process parameter variations (e.g. threshold voltage of a transistor caused by doping variation) are not local to a single transistor but are global, affecting multiple transistors. In this case, the "faulty group" could be obtained by grouping the affected components by their position on a die or by their component type (e.g. NMOS vs. PMOS). In a similar manner as in the single fault case, in the group fault model case simultaneously shifting all effected components out of their tolerance range performs fault injection.

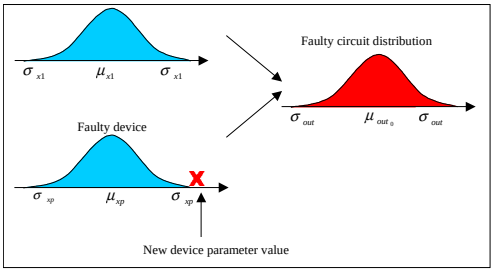

Figure 3: Faulty circuit distribution

7.2. Fault simulation

Similarly to the fault-free circuit case, instead of using one Monte Carlo simulation, we use statistical approach and linear estimation of the fault effect to obtain the faulty circuit PDF.

For each fault in the fault list, replacing the fault-free device statistical data by its faulty value (as discussed in the previous paragraph) performs fault injection.
Under this assumption, equation (2) and equation (3) become

$$\mu_{out}\big|_{X_p is\ faulty} = \mu_{out_0} + \sum_{i=1}^{N} S_{x_i}^{out}\mu_{\Delta x_i}$$
$$= \mu_{out_0} + \sum_{i=1, i\notin p}^{N} S_{x_i}^{out}\mu_{\Delta x_i} + \sum_{i=1, i\in p}^{N} S_{x_i}^{out}(\mu_{\Delta x_i} + 3\sigma_{x_i})$$
$$= \mu_{out_0} + \sum_{i=1, i\in p}^{N} S_{x_i}^{out}(3\sigma_{x_i}) \qquad \textbf{(4)}$$

And

$$\sigma^2_{out}\big|_{X_p is\ faulty} = \sum_{i=1, i\notin p}^{N} (S_{x_i}^{out})^2 \sigma^2_{x_i} + \sum_{i=1, i\notin p}^{N}\ \sum_{j=1, j\neq i\notin p}^{N} S_{x_i}^{out} S_{x_j}^{out} \sigma_{x_i x_j} \qquad \textbf{(5)}$$

Where $\mu_{out}\big|_{X_p is\ faulty}$ and $\sigma^2_{out}\big|_{X_p is\ faulty}$ represent the mean and the standard deviation of the faulty circuit respectively. "p" is the set of faulty elements under test: this set contains either one component in the single fault model case or all components of the "faulty group" in the group fault model.

Note that in this paper only the single fault model is implemented. The extension to "group fault" model is strait forward.

8. Measurement of circuit testability

Due to the continuous nature of analog circuit, the distinction between the fault-free circuit and the faulty circuit is not as clear as in the digital case where stuck-at fault model effect is seen as a 0/1 effect on the output.
Figure 4 shows a typical digital case where there could be no error in the distinguishing between the fault-free and the faulty circuit. On the other hand, *Figure 5* shows a typical analog case where there is no clear distinction between the fault-free and the faulty circuit. From *Figure 5* it is clear that if any decision is made regarding the circuit, this decision is subject to error. There is a need for a way to be able to measure the error that is committed and thus to measure the degree of testability of a circuit. Once the "degree of testability" is obtained, circuit classification could be done.

Let set up the two hypothesis:

- $\mathcal{H}_0$: the circuit is faulty and
- $\mathcal{H}_1$ the circuit is fault-free

The decision to accept or reject any hypothesis is subject to error. Two kinds of errors may be made. If the circuit is accepted as fault-free circuit when it is faulty then a *type I error* occurred. If the circuit is rejected as faulty circuit when it is fault-free a *type II error* is present. The probability of occurrence of type I and type II errors are thus defined as $P(\mathcal{H}_1|\mathcal{H}_0)$ and $P(\mathcal{H}_0|\mathcal{H}_1)$ respectively.

The notation $P(\mathcal{H}_i \mid \mathcal{H}_j)$ indicates the probability of deciding $\mathcal{H}_i$ when $\mathcal{H}_j$ is true. In the literature $P(\mathcal{H}_1 \mid \mathcal{H}_0)$ is commonly referred to as the probability of false accept (*PFA*) and $P(\mathcal{H}_0 \mid \mathcal{H}_1)$ is referred to as the probability of false reject (*PFR*). These errors are illustrated in *Figure 5*. These two errors are unavoidable to some extent and *it is not possible to reduce both error probabilities simultaneously*.

However, those errors may be traded off against each other. To do so we need only to change the threshold as shown in *Figure 6*. Clearly, the type I error probability is decreased at the expense of increasing the type II error probability. Trading one probability against the other, change the circuit's quantification mechanism and could have serious repercussion: if we decide the device parameter is fault-free but it proves to be defective, the entire circuit could be defective and we incur a large cost (packaging, additional testing...). If however, we decide the device parameter is defective when it is not, we incur the smaller cost of the circuit only. Thus, care should be considered when selecting a new threshold depending on the circuit at hand and the type of test that the user is interest in.

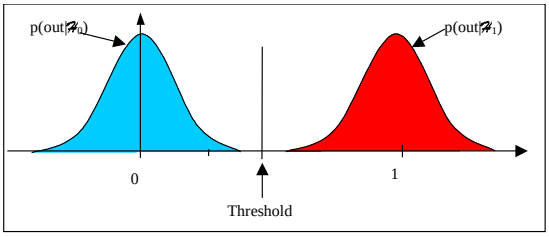

Figure 4: Digital circuit fault-free and faulty probability distribution function

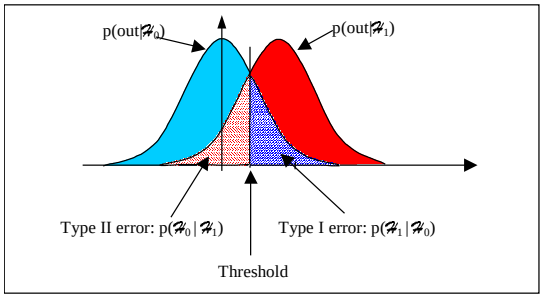

Figure 5: Possible hypothesis testing errors and their probabilities.

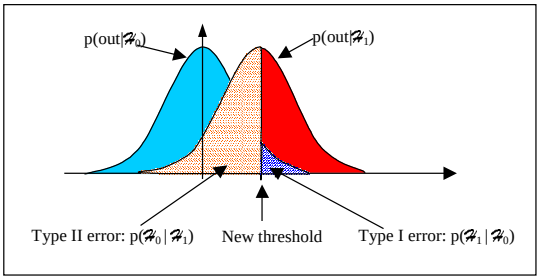

Figure 6: Trading off errors by adjusting threshold

Probability of false accept and probability of false reject are the natural way to represent the error in each one of the two hypothesis. False accept and false reject will be used as criteria for analog circuit test vector generation.

9. Test vector generation

Now all the building blocks are introduced to allow us to generate analog test vectors. First the fault-free and the faulty circuit PDF's functions are obtained by using the mapping of the device statistical data and the fault effect to the output parameter. Then, the obtained probability of false rejects and false accept are used as a method to quantify the classification error.

Now those building blocks will be used to identify the best stimulus that minimizes the probability of taking the wrong decision on the circuit. In another words we want to identify the stimuli that minimizes the "*Bayes risk*" defined as

$$R = \sum_{i=0}^{1}\sum_{j=0}^{1} C_{ij} P(H_i / H_j) P(H_j) \qquad \textbf{(6)}$$

Where C_{ij} is the cost if we decide $\mathcal{H}_i$ but $\mathcal{H}_j$ is true. $P(\mathcal{H}_i \mid \mathcal{H}_j)$ the probability of deciding $\mathcal{H}_i$ when $\mathcal{H}_j$ is true and $P(\mathcal{H}_j)$ the probability of occurrence of the fault. Usually if no error is made, we do not assign a cost so that $C_{00} = C_{11} = 0$. In addition since each fault is tested independently of the other faults and the probability of the fault being present in the circuit is independent of the applied stimulus, then, the probability of occurrence of the fault $P(\mathcal{H}_j)$ could be set to a constant, namely 1. Under those assumptions equation (6) is reduced to

$$R = C_{10} P(H_1 / H_0) P(H_0) + C_{01} P(H_0 / H_1) P(H_1) \qquad \textbf{(7)}$$

The Bayes risk *R* is computed for all stimulus and for each fault in the fault list. The stimuli for which *R* is minimal, is taken as the test vector for the fault under consideration

$$\text{Test Vector} = \text{Stimuli} \mid R(\text{stimuli}) \text{ is minimal} \qquad \textbf{(8)}$$

For example, lets consider a simple case where the possible stimuli are a predefined frequency range [f1, f2]. For each frequency in [f1,f2] the fault free circuit and the faulty circuit PDF functions are computed for each device parameters in the fault list. The test vector is the stimuli for which the two distributions are further apart (*Figure 7*).

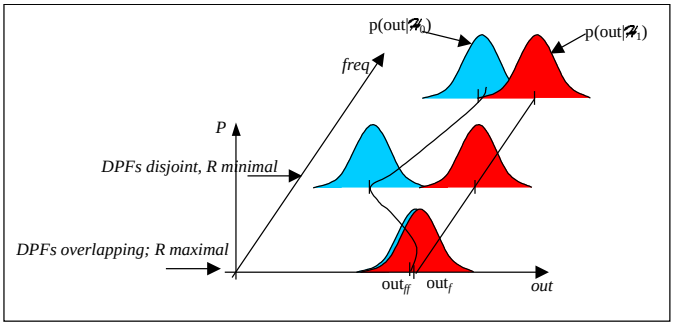

Figure 7: Fault free circuit PDF and faulty circuit PDF as function of the frequency

10. Experimental results

The set of 7 benchmark circuits have been simulated. The benchmark circuits range from an operational amplifier to a 4 bit current DAC. Level 3, Level 28 and level 49 transistor MOSFET models are used.

All the principal components (resistances, capacitances, and transistors W and L) where considered in the fault dictionary with 5% statistical variation. The output voltage and the output current (including IDDQ) were considered for testing. The output detection threshold for false accept and false reject computation was set to $3\sigma_{out}$ (the output standard deviation of the fault-free circuit) such that $PFR \approx 0$ and $fault\ coverage = 1 - R \approx 1 - PFA$.

Circuit statistics and simulation results are summarized in *Table 1*. This table gives: the circuit name, the number of faults injected in the circuit, the obtained fault coverage, the number of test vectors used to obtain the indicated fault coverage, and their corresponding test domain.

Note that stimulus in the AC, DC and transient domains have been used for testing. The combination of these domains is also possible.

It is known that soft faults are very difficult to detect mainly due to the feedback loops and to the nature of the analog signals. From the test results, soft faults could be classified into two different types: the internal faults (inside closed loop Op Amps) and external faults.

External faults are relatively easy to test since they have direct impact on the circuit outputs. The state variable filter circuit and the low pass filter circuit are perfect examples of external faults where fault coverage is 100% and 80% respectively.

Internal faults are much more difficult to detect. The Chebychev filter is good example of low fault coverage (35%). This low coverage is mainly due to the low

sensitivity of the output gain with respect to the transistors width and length.

In an attempt to increase fault coverage, the unity gain closed loop OpAmp configuration has been changed to an open loop. This change increased fault coverage from 63% to 91%. However, we should be very careful while reading those results, since for circuits with high nonlinearity and with high gain, a first order sensitivity equations may not be sufficient to adequately describe the circuit behavior.

In another attempt to increase fault coverage, the stimulus space has been increased by adding new test vectors. IDD testing proved to be a major supplement to the voltage testing: In the Unity gain OpAmp case, fault coverage increased from 63% to 83% and in the kfiltre case fault coverage increased from 48% to 63%.

Despite the difficulties in defining and testing soft faults, our methodology efficiently simulated faults and generated test stimulus for the deterministic fault coverage. Indeed in the unity gain OpAmp detailed comparison between Hspice MonteCarlo results and equation (4) and (5) showed that the average error on the faulty output mean value is 2%., and the average error on the faulty output standard deviation value is 14%

Table 1: Fault Simulation and Test Vector Result Summary

Circuit	No of faults	Coverage	Test Vectors
Unity gain OpAmp	44	63% No IDD 83% with IDD	DC (v): 9.9, -4.9, 4.8
Open loop OpAmp	42	91%	DC (v): 9.9, -0.2, 0.0, 0.1, 0.3
Kfiltre	58	48% No IDD 63% with IDD	AC (Hz): 165k, 229k, 239k, 138k, 0.186meg, DC (v): –10.0 v
I_DAC 4bits	50	44%	Transient (input code): 1110, 1101, 1011, 0111, 0011
Low pass filter	10	80%	Transient (s): 70u, 0.138m, 0.141m
Chebyche v filtre	150	35%	AC (Hz): 15.1k
State variable filter	26	100%	AC (Hz): 1.34, 5.75k , 6.16k, 6.45k, 97.7k

11. Conclusion

In this paper we have presented a method for fast and accurate simulation of the fault-free and the faulty circuits. Indeed, for each fault in the fault dictionary and for all possible stimulus of a circuit, a faulty circuit PDF function is generated. This process (generation of the PDF functions) relies on the process information and on the output sensitivities to the circuit principal components in order to generates statistical models of the fault-free and the faulty circuits (one for each fault in the fault dictionary). Note that since no physical Monte Carlo simulation where performed, the fault simulation time was only a fraction of a full Hspice Monte Carlo simulation without a significant degradation of the accuracy of the results.

Once the fault-free circuit distribution and the faulty circuit distribution where obtained for each fault in the circuit, we compute the false accept for the entire range of operation. The test vector is defined as the stimuli for which the false accepts is minimal. The same methodology is applied for false reject test vector generation.

This method has been implemented as the automated tool set for fault simulation and test vector generations called FaultMaxx and TestMaxx respectively. The tools are extended by hard fault modeling, simulation and test vector generation and by comprehensive trade-off and compaction capabilities. The technical aspects of these extensions are subject to other publication.

Reference

[1] Analytical Device Modeling for MOS Analog IC's Based on Regularization and Baysesian Estimation, M.Conti, S. Orcioni, C. Turchetti, G. Soncini, and N. Zorzi. IEEE Transaction on computer-aided design of integrated circuits and systems. Vol. 15, No. 11, Nov. 1996.

[2] Ben-Hamida N. and Kaminska B., "Analog Circuit Testing Based on Sensitivity Computation", IEEE International Test Conference, Baltimore, Oct. 1993, pp. 652-661.

[3] K. Saab, D. Marche, Ben-Hamida N., B. Kaminska, "LIMSoft: Automated Tool For Sensitivity Analysis and Test Vector Generation", IEE Proceeding on Circuit Devices and Systems, Vol. 143, No. 6, Dec. 1996.

[4] Slamani M., Kaminska B., "Testing Analog Circuit by Sensitivity Computation", European Design Automation Conference, EDAC'92, Bruxelles, March 1992, pp. 563-569.

[5] Slamani M., Kaminska B.,"Multifrequncy Testability Analysis for Analog Circuit", IEEE Trans. on Circuit and Systems, part II, feb 1996, pp. 134-139.

[6] Ben-Hamida N., Saab K., Marche D. and Kaminska B.,"FaultMaxx: A Perturbation Based Fault Modeling and Simulation for Mixed-Signal Circuits", Asien Test Conference. October 1997.

[7] Manoj Sachdev, " A Realistic Defect Oriented Testability Methodology for Analog Circuits", Jetta 1994.

[8] Naveena Nagi, Abhijit Cahtterjee, Jacob Abraham, "Fault Simulation of Linear Analog Circuits", Analog Integrated circuits and Signal Processing, 1993, pp. 245-260.

[9] P. Yang, D.E. Hocevar, P.F. Cox, C. Machala, P.K. Chattergee. "An Integrated and Efficient Approach for MOS VLSI Statistical Circuit Design". IEEE Transaction on computer-aided design of integrated circuits and systems. Vol. CAD 5, No. 1, Jan. 1986.