Journal of the Brazilian Computer Society

versão impressa ISSN 0104-6500

J. Braz. Comp. Soc. v. 4 n. 2 Campinas Nov. 1997

http://dx.doi.org/10.1590/S0104-65001997000300001

A CAT Tool for Frequency-domain Testing and Diagnosis on Analog

Érika Fernandes Cota
erika@inf.ufrgs.br
CPGCC/DELET
Universidade Federal do Rio Grande do Sul
Av. Osvaldo Aranha, 103 CEP 90035-190
Porto Alegre — RS — Brasil

Elias José Di Domênico
erika@inf.ufrgs.br
CPGCC/DELET
Universidade Federal do Rio Grande do Sul
Av. Osvaldo Aranha, 103 CEP 90035-190
Porto Alegre — RS — Brasil

Marcelo Lubaszewski
erika@inf.ufrgs.br
CPGCC/DELET
Universidade Federal do Rio Grande do Sul
Av. Osvaldo Aranha, 103 CEP 90035-190
Porto Alegre — RS — Brasil

Abstract *This paper presents a sensitivity-based test generation tool for analog multifrequency testing and diagnosis. The test generation procedure is based on sensitivity analysis and on fault simulation. This tool generates minimal test sets that maximize the coverage of soft, large and hard component faults and that enhance the coverage of interconnect shorts. An introduction to the problem of analog fault diagnosis considering both component and interconnect faults, is also presented. This procedure is now being automated by integrating commercially available tools for symbolic computation and electrical simulation.*

Keywords: *Computer-Aided Testing, Automatic Test Generation, Analog and Mixed-signal Circuits*

1 Introduction

Mixed-signal integrated circuits have been widely used in automotive and medical applications, in nuclear and space systems, in industrial automation, etc. Since these applications require some interaction between the analog and the digital world, several technologies have emerged that allow the implementation of both circuitry types on the same substrate. In the same proportion, the development of testing technologies have become a need, specially for the analog parts included in these devices, since practical analog testing solutions are still lagging behind their digital counterparts. As a consequence, techniques that can help in the automation of analog testing are nowadays of much interest.

Besides testing, fault diagnosis becomes specially important for prototype debugging and, during the circuit lifetime, whenever repair is possible. Nevertheless, the diagnosis problem can be very hard to solve, since many faults may produce an identical behavior at the circuit outputs.

Within this context, this work faces the problem of automatically generating multifrequency tests for analog circuits. The test methodology is based on sensitivity analysis and generates detection ranges for soft, large and hard faults in passive components. Shorts between two nodes (interconnect faults) are also considered in the fault model. From the detection ranges generated by the test methodology, the diagnosis procedure is developed. The ultimate goal is to generate a minimal test set that guarantees maximal diagnosis and maximal coverage of all possible faults.

2 State-of-the-Art and Contributions

Some recent papers have been addressing the analog testing problem and interesting solutions and methods have emerged that optimize the test process.

The studies that are of interest to this work deal with three major topics: fault modeling and test methods based on fault simulation; test methods based on sensitivity analysis and ATPG for fault detection and/or fault diagnosis.

In the first group, the works define models for those faults that can occur in analog circuits. For instance, [1] presents models for parametric and catastrophic faults in passive components, [10] introduces large deviations of passive components, and [4] defines fault models for injecting shorts between digital and analog parts into the circuit description. In terms of test generation, the procedure proposed in [1] is based on the fault modeling introduced in [2], while the process of searching input stimuli of [8] considers catastrophic and parametric, AC and DC faults in passive and active components [12] and is assisted by a fault simulator [13].

In the second group, authors investigate the circuit faulty behavior by using the transfer function and the sensitivity concept. This means that deviations are observed at the primary outputs as a response to some variation in a circuit component. Some works analyze the circuit testability in a functional way [2], others work directly with the sensitivity concept using a symbolic approach [3, 10].The work presented in [3], for example, defines necessary and sufficient conditions to choose diagnosis parameters. The one presented in [10] chooses test frequencies to detect soft/large deviations and catastrophic faults in a multifrequency analysis. However, the test set generated is not minimum, since they try to combine specific frequency values rather than frequency ranges.

In general, the papers addressing the diagnosis pro-blem present methodologies to optmize the test process, by defining a minimal test set that can lead to the location of the faults considered in the model. Usually, diagnosis tools follow two basic approaches [11]: Simulation Before Test (SBT) and Simulation After Test (SAT). In the SBT method the design is analyzed before testing, and the output of a faulty circuit is stored in a fault dictionary, as in [7]. Thus, if a fault is detected, it can be located by matching the actual result against the dictionary. The SAT approach, on the other hand, makes a mathematical analysis, based on diagnosis equations, following the detection of a fault.

In this paper, a new test generation procedure is proposed which enlarges the set of analog faults considered and merges the main features of some of the previous works mentioned above: the sensitivity analysis of soft faults, large deviations and hard faults [10], the search of minimal sets of test measures and test frequencies [7], and the generation of tests for interaction shorts based on fault simulation [8]. Considering the diagnosis problem, we extended the SBT method proposed in [7] in order to consider interconnect faults and multiple measures. Based on these four axes, an automatic test generation tool is presented here that makes use of commercially available software (SSpice, Maple, HSpice and SIS) in association with some in-house made tools.

3 Preliminaries

In order to detect faults in analog circuits, it is necessary to define a valid range for the output parameters. This range must be defined by the designer so that any output value within it is considered correct. These values are related to each design and to the accuracy of the components and the test equipment in use. Test parameters can be associated to the primary outputs or to some internal nodes of the circuit. We consider herein the former.

3.1 Fault modeling

The fault model used includes soft and large passive component deviations, and hard faults [10]. Shorts between pairs of nodes (interaction faults) were modeled in [4] and are also addressed here. Faults internal to operational amplifiers [12] are not considered in this paper.

Soft faults [2] are small deviations around the nominal value of a component. They may cause a circuit malfunction such as a change in the cut-off frequency of a filter, in the output gain of an amplifier, etc.

Large deviations [10] are also deviations in the nominal value of components, but of a greater magnitude (in general around 50%). They still cause circuit malfunctions, but their effects are quite different from those observed for soft faults.

Hard faults [3] are serious changes in component values.These faults can even modify the structure of the design and are usually due to interconnect or component open and short circuits.

A type of fault that often occurs in board manufacturing technologies is shorts between nodes of the circuit. This kind of faults was modeled in [4] but was not considered in previous automatic test generation tools. Because its effect is strongly dependent on the circuit topology, there is no methodology test that ensures its detection.Figure 1 shows an example of this fault in an analog block. In this work, pairwise shorts involving circuit nodes that are not terminals of the same component are dealt with.

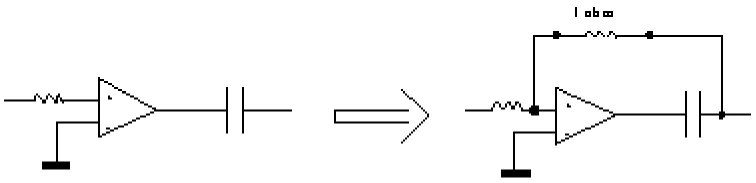

Figure 1: Analog interaction fault modeled as a resistive short of 1Ω

3.2 Sensitivity computation

The circuit sensitivity is defined as the effect on a performance parameter T_j of a deviation in an element x_i of the circuit.This relative deviation can be expressed in different ways depending on its magnitude. The **differential sensitivity** is applied to small deviations (soft faults) only, while the **incremental**

sensitivity is used for small and large deviations (including hard faults). Then, considering an output parameter T_j denoted as $\frac{N(x)}{D(x)}$ and an element x_i, we have:

1. **Differential Sensitivity:** $S_{x_i}^{T_j} = \frac{x_i}{T_j} \frac{\partial T_j}{\partial x_i}$ **(1)**

2. **Incremental sensitivity:** $\wp_{x_i}^{T_j} = \frac{S_{x_i}^{T_j}}{1 + S_{x_i}^{D} \frac{\Delta x_i}{x_i}}$ **(2)**

where $S_{x_i}^{D}$ is the differential sensitivity of the denominator of T_j.

A sensitivity analysis can be made experimentally [1] or symbolically [10]. Herein, symbolic calculations are used. This way, it is possible to optimize the test generation process and obtain more information about a faulty analog circuit.

4 Test generation procedure

Figure 2 outlines the test generation procedure proposed in this work. Basically, the procedure consists of a structural circuit analysis starting from the definition of its transfer function (this allows to make a symbolic sensitivity analysis). After the definition of the transfer function and its sensitivity analysis, a minimal set of parameters and input signals that guarantee maximal fault coverage is computed. This set must also consider interaction faults, so that these faults can be detected just as component faults are.

Step 1: Once the transfer function is defined, the symbolic analysis of the sensitivity is carried out by means of equation (2).

Step 2: After the sensitivity computation, three variables defined by the designer must be considered:

2.a) the test parameter **tolerance**, that corresponds to the maximal output deviation that is still considered correct. This information depends not only on the circuit specification, but also on the accuracy of the test equipment that will be used to measure the parameters.

2.b) the minimal **deviation** to detect in each component. This depends on the accuracy of the components used in the design.

2.c) the **range** of input signals to be considered.

Then, two fault detection information are made available to the test engineer: the minimal sensitivity from which the defined deviation will be detected, and the minimal deviation of each component that can be detected by the defined tolerance. With this, the test generation tool can show to the designer both the input ranges to be used to detect deviations above the minimal value and the elements or faults that are outside the detection regions.

Step 3: The minimal set of detection parameters and input signals is computed as follows:

3.a) For each test parameter and each component, a sensitivity-frequency curve is plotted for each kind of fault (soft, large, open and short). The resulting set of curves brings the information about the input ranges that detect every fault on a given component, considering deviations above and below the nominal values.

3.b) In each set of curves, the most restrictive range (or set of ranges) is taken so that it is ensured that all faults on that component are detected.

3.c) Properly combining (per test parameter) the set of ranges given by all component curves, a minimal set of test signals can be chosen so that maximal detection is achieved.

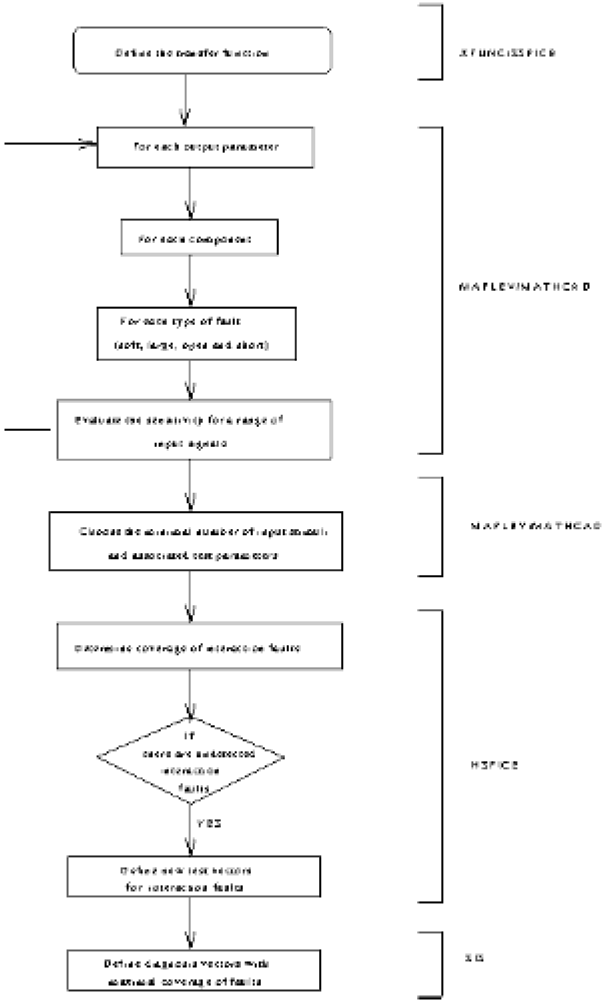
Define the transfer function
XFUNC/SSPICE
For each output parameter
For each component
For each type of fault
(soft, large, open and short)
Evaluate the sensitivity for a range of
input signals
MAPLE V/MATHCAD
Choose the minimal number of input stimuli
and associated test parameters
MAPLE V/MATHCAD
Determine coverage of interaction faults
If
there are undetected
interaction
faults
YES
Define new test vectors
for interaction faults
HSPICE
Define diagnosis vectors with
maximal coverage of faults
SIS

Figure 2: Proposed test generation methodology

Step 4: Then, the minimal deviations actually detected for each component are computed for the test signals obtained in the previous step.

Step 5: The interaction faults are dealt with in this step. The ranges defined in step 3.c are used to simulate these faults and evaluate their coverage. For those interaction faults not detected by the existing test signals, new input test stimuli are selected from the comparison of the electrical simulation of the faulty circuit to the electrical simulation of the fault-free circuit along the frequency range of interest.

Step 6: This step deals with the problem of diagnosis. The detection ranges defined above are reviewed in order to isolate each fault. The minimal set of diagnosis ranges that achieves maximal fault coverage is computed.

5 Experimental results

In order to validate the procedure discussed in section 3, it was applied to the biquad filter presented in figure 3.

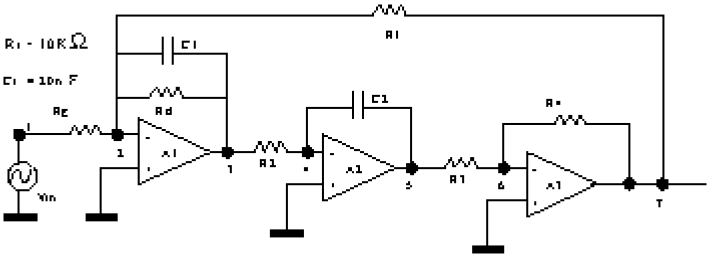

Figure 3: Biquadratic filter

5.1 Transfer function definition

The frequency-domain parameters that can be observed from the filter primary output are the gain and phase associated to node 7. The gain (G) and the phase (Φ)parameters are defined as:

$$G = \sqrt{\left(\frac{|R_1 R_4 R_d|}{\left|\left(4R3\,R2\,C2\pi^4 f^2\,R1RdC1 - 2j\,R3R2C2\pi fR1 - R4Rd\right)Rg\right|}\right)} \quad \textbf{(3)}$$

$$\Phi = arctan\left(\frac{1}{2}\,\frac{4R_g\,R3R2C2\pi^2 f^2\,R1RdC1 - R_g\,R4Rd}{R_g\,R3\,R2C2\pi fR1}\right) \quad \textbf{(4)}$$

These equations were obtained from the transfer function by the relations $G = |Ft|$ and $\Phi = arctan\left(\frac{b}{a}\right)$, where Ft is the transfer function, b is the complex part and a is the real part of the transfer function.

5.2 Sensitivity computation

Using the definitions presented in section 2.2 (equation 2) and the gain and phase equations (equations 3 and 4), the computation of the sensitivity was symbolically performed using a powerful tool for algebraic manipulation [6].

The parameter tolerances and minimal component deviations were defined as 5% (steps 2.a and 2.b in section 3) and the input range (frequency range) as [0Hz .. 5000Hz] (step 2.c).

5.3 Selection of the minimal set of parameters and input signals

The incremental sensitivity curves were plotted for each test parameter and each component considering the predefined frequency range. Two gain sensitivity curves for Rd are shown in figure 4.

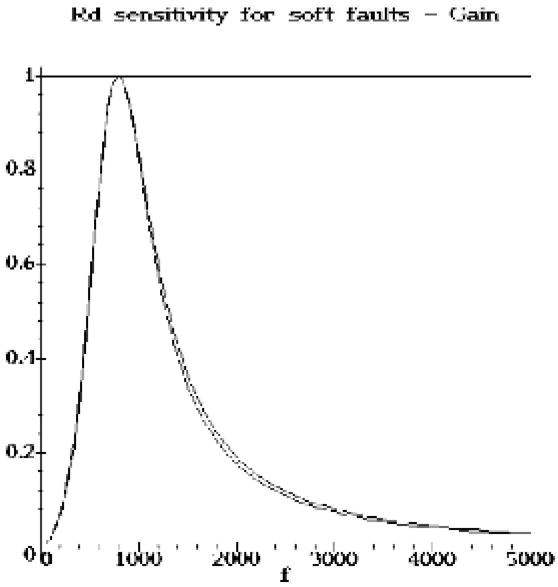

Figure 4a: Gain sensitivity for Rd soft deviations and for an open circuit fault.

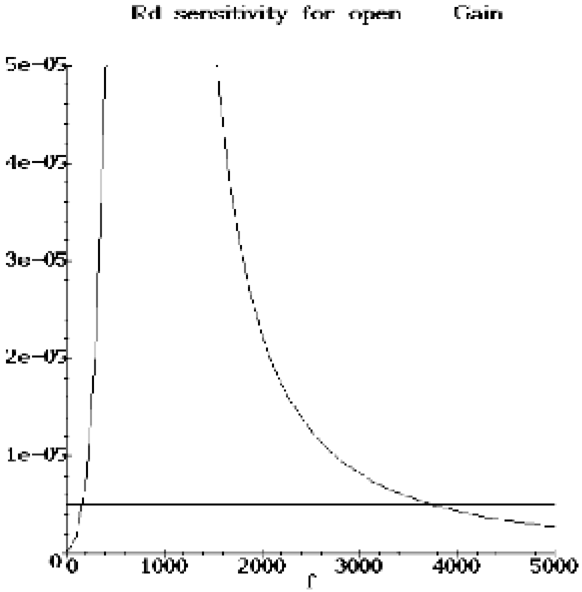

Figure 4b: Gain sensitivity for Rd soft deviations and for an open circuit fault.

These curves include a straight line corresponding to the minimal sensitivity value that ensures the detection of the component deviations. This value is given by the expression $sens_{min} = \frac{tolerance}{deviation}$ (steps 2.a and 2.b).

From all the curves plotted for all components, the test frequency ranges were computed and are presented in tables 1 and 2 for the parameters gain and phase.

Component	Soft Faults	Large Deviations	Hard Faults	
			Short	Open
Rg	[0..5000]	[0..5000]	[0..5000]	[0..5000]
C1	[1825..5000]	[270..745] or [845..5000]	[1.5..5000]	[192..5000]
Rd	—	[300..2200]	[1.5..5000]	[200..3724]
R1	—	[0..737] or [887..1925]	[0..5000]	[0..3438]
R2	[850..3375]	[437..5000]	[362..5000]	[35..5000]
R3	[850..3362]	[437..5000]	[355..5000]	[35..5000]
R4	[825..5000]	[487..5000]	[1.6..5000]	[375..5000]
C2	[850..3400]	[440..5000]	[45..5000]	[361..5000]

Table 1: Test frequency ranges to detect each kind of fault using the test parameter gain

From these tables, the following can be concluded:

1 Deviations in Rg will affect only the test parameter gain. No fault in this component generated a phase sensitivity curve where a frequency range could be selected.

2 For the phase parameter, the minimal sensitivity necessary to detect soft faults was not achieved for any component. This indicates that the design is robust for this parameter, what means that only deviations above the components accuracy will be signaled as actual faults. Looking at the gain parameter, only Rd and R1 have this feature.

Then, the minimal set of frequency ranges that guarantees maximal coverage is obtained from the ranges shown in tables 1 and 2 for the parameters gain and phase:

Gain: [1825..1925]Hz

Phase: [475..722]Hz or [865..2512]Hz consi-dering that faults in Rg are not detected.

Component	Soft Faults	Large Deviations	Hard Faults	
			Short	Open
Rg	—	—	—	—
C1	—	[475..5000]	[2.75..5000]	[412..5000]
Rd	—	[240..722] or [865..5000]	[130..755] or [832..5000]	[0..5000]
R1	—	[220..2525]	[127..3500]	[0..5000]
R2	—	[212..2525]	[125..3520]	[0..5000]
R3	—	[220..2525]	[125..3525]	[0..5000]
R4	—	[232..2520]	[125..3488]	[0..5000]
C2	—	[222..2512]	[125..3475]	[0..5000]

Table 2: Test frequency ranges to detect each kind of fault using the test parameter phase

For this example and for the gain parameter, a single input frequency is sufficient to detect faults in all components. For the phase parameter, a single frequency can also be chosen, but faults in Rg will never be detected. This means that the gain is the best test parameter and any frequency within the selected range can be used as a test frequency. For the sake of simplicity, the test generation tool chooses a frequency in the middle of the range.

Other circuits may need more input stimuli to ensure maximal detection of faults. Since it is desirable to decrease the number of input signals necessary to test, the gain is the parameter selected for fault detection in the filter components.

The next step consists of computing the minimal deviations actually detected by each parameter at the circuit output. An information that comes along is what faults cannot be detected by this method. The sensitivity values of each parameter for every component in the boundaries of the test frequency ranges are computed. Table 3 shows the minimal deviations obtained for the test parameter gain.

From table 3 the following remarks can be made:

1 The elements that do not have a detection range in table 1 present in table 3 a minimal deviation greater than the minimum defined by the designer (5%).

2 For the other components checked by the gain parameter, deviations smaller than 5% will be detected and labeled as faults. It means that the design is not sufficiently robust when considering these elements and this parameter. All these data are made available to the designer by the test generation tool and he can thus choose the best solution: either change the design, or keep the test procedure despite of the possible false rejection of parts.

Component	Nominal Value	1825Hz	1875Hz	1925Hz
Rg	10	-4.76%	-4.76%	-4.76%
		+5.26%	+5.26%	+5.26%
C1	20^{-9}	-4.97%	-4.95%	-4.93%
		+5.50%	+5.47%	+5.45%
Rd	10K	-18.98%	-19.94%	-20.91%
		+26.89%	+29.09%	+31.43%
R1	10K	-20.74%	-21.56%	-22.39%
		+40.68%	+43.63%	+46.81%
R2	10K	-4.03%	-4.06%	-4.09%
		+4.45%	+4.49%	+4.52%
R3	10K	-4.03%	-4.06%	-4.09%
		+4.45%	+4.49%	+4.52%
R4	10K	-4.26%	-4.29%	-4.32%
		+4.20%	+4.23%	+4.26%
C2	20^{-9}	-4.03%	-4.06%	-4.09%
		+4.45%	+4.49%	+4.52%

Table 3: Minimal component deviations detected by the filter gain

5.4 Coverage of interaction faults

From tables 1 and 2 it is possible to get a minimal set of detection ranges and test parameters for all components. The next step consists of determining the coverage of interconnect faults using the test vectors defined for components. To do this, electrical simulations were performed using HSpice and, for the undetected faults, new detection ranges were determined. The results of these simulations are presented in table 4.

The first column indicates the fault simulated (1-3 is a short between nodes 1 and 3) and the second one indicates if it is detected by one of the test parameters or not.

From table 4, it can be noted that the coverage of interaction faults for the test parameter and test frequency chosen (Gain at 1875Hz) is 10 over 14 possible faults (71,4%).

New ranges allowing the detection of the four undetected faults were determined by comparing the electrical simulation of the faulty and the fault-free circuit along the whole frequency range.

The individual new ranges are shown in the third and fourth columns of table 4. Symbol ∞denotes values above 100KHz. By intersecting these individual ranges, one obtains the following global ranges for the gain and the phase parameters:

Gain: [24.5K..∞] Hz

Phase: [1.48K..∞] Hz

Fault	Detection status	New gain range	New phase range
1-3	detected	—	—
1-4	detected	—	—
1-5	detected	—	—
1-6	detected	—	—
1-7	detected	—	—
2-4	detected	—	—
2-5	detected	—	—
2-6	undetected	[490..1.86k] or [2k..∞]	[147.. ∞]
3-5	undetected	[3k.. ∞]	[1.48k.. ∞]
3-6	detected	—	—
3-7	undetected	[0..1.8k] or [2k..∞]	[758..∞]
4-6	detected	—	—
4-7	detected	—	—
5-7	undetected	[24.5k..∞]	[1.1k..∞]

Table 4: Detection ranges for interaction faults for each test parameter

Note that the new range obtained for the gain requires very high frequencies,for which the output gain is extremely low, since the filter has a low-pass function. This way, the test parameter to choose for improving the detection of interaction faults is the phase.

Then, the parameters to observe and the final detection ranges for this circuit are:

Gain: [1825..1925], for the detection of component faults.

Phase: [1.48K..5K], for the detection of interaction faults.

6 Diagnosis of Component Faults

Recombining the detection ranges in tables 1 and 2, as shown in tables 5 (for gain) and 6 (for phase), the sub-ranges where component faults are detected can be identified. The boundaries of each detection range in tables 1 and 2 will form a new range in tables 5 and 6.

Range	Rg	C1	Rd	R1	R2	R3	R4	C2
A1:[0..300]	1	0	0	1	0	0	0	0
A2: [300..737]	1	0	1	1	0	0	0	0
A3: [737..825]	1	0	1	0	0	0	0	0
A4: [825..850]	1	0	1	0	0	0	1	0
A5: [850..887]	1	0	1	0	1	1	1	1
A6: [887..1825]	1	0	1	1	1	1	1	1
A7: [1825..1925]	1	1	1	1	1	1	1	1
A8: [1925..2200]	1	1	1	0	1	1	1	1
A9: [2200..3362]	1	1	0	0	1	1	1	1
A10: [3362..3375]	1	1	0	0	1	0	1	1
A11: [3375..3400]	1	1	0	0	0	0	1	1
A12: [3400..5000]	1	1	0	0	0	0	1	0

Table 5: Diagnosis ranges for the gain parameter

For these tables, columns indicate whether a diagnosis range is included in the detection range of an element (1) or not (0). Equivalent faults are defined as faults that are detected by exactly the same ranges. These faults cannot be distinguished by this method and are considered as a single fault. For this example, there are no equivalent faults for any parameter. As explained in [7], a boolean function can be extracted from these tables, in the form of product of sums. A sum term is defined for each group of two components and consists of the ranges that detect faults in just one of them. Below this extraction procedure is applied to table 5.

Range	Rg	C1	Rd	R1	R2	R3	R4	C2
B1: [0..212]	0	0	0	0	0	0	0	0
B2: [212..220]	0	0	0	0	1	0	0	0
B3: [220..222]	0	0	0	1	1	1	0	0
B4: [222..232]	0	0	0	1	1	1	0	1
B5: [232..240]	0	0	0	1	1	1	1	1
B6: [240..475]	0	0	1	1	1	1	1	1
B7: [475..722]	0	1	1	1	1	1	1	1
B8: [722..865]	0	1	0	1	1	1	1	1
B9: [865..2512]	0	1	1	1	1	1	1	1
B10: [2512..2520]	0	1	1	1	0	1	1	0
B11: [2520..2525]	0	1	1	0	0	1	0	0
B12: [2525..5000]	0	1	1	0	0	0	0	0

Table 6: Diagnosis ranges for the phase parameter

Z = (A1 + A2 + A3 + A4 + A5 + A6) * (A1 + A9 + A10 + A11 + A12)*
(A3 + A4 + A5 + A9 + A10 + A11 +A12) * (A1 + A2 + A3 + A4 + A11 + A12)*

(A1 + A2 + A3 + A4 + A10 + A11 + A12) * (A1 + A2 + A3)*
(A1 + A2 + A3 + A4 + A12) * (A2 + A3 + A4 + A5 + A6 + A9 + A10 + A11 + A12)*
(A1 + A2 + A6 + A8 + A9 +A10 + A11 + A12) * (A5 + A6 + A11 + A12)*
(A5 + A6 + A10 + A11 + A12) * (A4 + A5 + A6) * (A5 + A6 + A12)*
(A1 + A3 + A4 + A5 + A8) * (A2 + A3 + A4 + A9 + A10)*
(A2 + A3 + A4 + A9) * (A2 + A3 + A9 + A10 + A11 + A12)*
(A2 + A3 + A4 + A9 + A10+ A11)*(A1 + A2 + A5 + A8 + A9 + A10)*
(A1 + A2 + A5 + A8 + A9) * (A1 + A2 + A4 + A5 + A8 + A9 + A10 + A11 + A12)*
(A1 + A2 + A5 + A8 + A9 + A10 + A11)*(A10)*(A4 + A11 + A12)*(A11)*
(A4 + A10 + A11 + A12) * (A10 + A11) * (A4 + A12); **(5)**

This equation is then simplified and any of the final product terms with the fewest literals is chosen as the minimal set of input signals that detect and diagnose component faults. For this example, equation 6 is the minimal form for equation 5.

Z = A3*A6*A9*A10*A11*A12 + A1*A6*A9*A10*A11*A12 +
+ A1*A5*A9*A10*A11*A12 + A3*A4*A9*A10*A11*A12 +
+ A3*A6*A8*A10*A11*A12 + A2*A6*A8*A10*A11*A12 +
+ A3*A4*A8*A10*A11*A12 + A2*A3*A6*A10*A11*A12 +
+ A1*A3*A6*A10*A11*A12 + A1*A2*A6*A10*A11*A12 +
+ A3*A4*A6*A9*A10*A11 + A3*A4*A6*A8*A10*A11 +
+ A3*A5*A10*A11*A12 + A2*A5*A10*A11*A12 +
+ A2*A4*A10*A11*A12 + A1*A4*A10*A11*A12 +
+ A2*A4*A6*A10*A11 + A1*A4*A6*A10*A11 +
+ A3*A4*A5*A10*A11 + A2*A4*A5*A10*A11 + A1*A4*A5*A10*A11 **(6)**

Any product term of this minimal form is a possible diagnosis set for the filter. The choice among them is made based on the number of literals and the maximal coverage of them. This way, terms with fewer literals are selected and the one with maximal fault coverage is taken as the final diagnosis set. Considering equation 6, there are 9 minimal terms, with 5 literals each. Unfortunately, some of them cannot detect faults in all components. This is the case of A3*A5*A10*A11*A12, that does not detect faults in R1. Three other sets are in this situation and the final set must be chosen among the remaining ones. Thus, to guarantee the diagnosis of component faults with maximal coverage, at least 5 test vectors must be applied and each vector belongs to one range selected from equation 6. If set A2*A4*A6*A10*A11 is chosen, for example, the final vectors will be 518.5Hz, 837.5Hz, 1356Hz, 3368.5Hz and 3387.5Hz, respectively.

This analysis was also made for the phase parameter. Equation is derived from table 6.

Z = (B7 + B8 + B9 + B10 + B11 + B12) * (B6 + B7 + B9 + B10 + B11 + B12)*
(B3 + B4 + B5 + B6 + B7 + B8 + B9 + B10) * (B2 + B3 + B4 + B5 + B6 + B7 + B8 + B9)*
(B5 + B6 + B7 + B8 + B9 + B10) * (B4 + B5 + B6 + B7 + B8 + B9)*(B6 + B8)*
(B3 + B4 + B5 + B6 + B11 + B12) * (B2 + B3 + B4 + B5 + B6 + B10 + B11 + B12)*
(B3 + B4 + B5 + B6 + B12) * (B5 + B6 + B11 + B12)*
(B4 + B5 + B6 + B10 + B11 + B12) * (B3 + B4 + B5 + B8 + B11 + B12)
(B2 + B3 + B4 + B5 + B8 + B10 + B11 + B12) * (B3 + B4 + B5 + B8 + B12)*
(B5 + B8 + B11 + B12) * (B4 + B5 + B8 + B10 + B11 + B12)*
(B2 + B10)*(B11)*(B3 + B4) * (B3 + B10) * (B2 + B10 + B11)*
(B2 + B3 + B4 + B10)*(B2 + B3) * (B3 + B4 + B11)*(B3 + B10 + B11)*(B4 + B10); **(7)**

The minimal form of this equation is:

Z = B2*B4*B8*B10*B11 + B2*B4*B6*B10*B11 +
+ B2*B3*B4*B8*B11 + B2*B3*B4*B6*B11 +
+ B3*B8*B10*B11 + B3*B6*B10*B11; **(8)**

There are 2 possible diagnosis sets in this case, each one with 4 detection ranges. Both of them can detect faults in all components (except for Rg, as previously mentioned). If B3*B8*B10*B11 is chosen, for instance, the test vectors will be 221Hz, 793.5Hz, 2516Hz and 2522.5Hz.

7 Diagnosis of Interconnect Faults

In the previous section was proposed a method to compute a minimal set of input signals that guarantees maximal diagnosis and maximal coverage of component faults. In this section, interconnect faults are considered using this method.

There are three possibilities to deal with diagnosis of interconnect faults for our case study. The first one is to consider both, component and interconnect faults, measuring the gain parameter. The second one is to measure the phase parameter, including the new ranges computed to detect interconnect faults with this parameter. Finally, the last approach is to consider both gain and phase parameters to detect all kinds of faults. This way, test vectors would have the form $Ai*Bi$, where Ai states for input signals for which the gain would be measured and Bi for those for which the phase would be observed.

7.1 Diagnosis by Gain Measurement

In section 6 it was shown that it is possible to generate a minimal set of input signals to detect and diagnose all component faults using the gain parameter. It is thus reasonable to think of diagnosing interconnect faults using this parameter. Then, table 5 is modified to include the possible interconnect faults and the new ranges necessary to detect them (given in table 4). The result is a matrix with 35 detection ranges and 25 faults. Although some equivalent faults are eliminated, the dimensions of this matrix are considerably larger than in table 5 and the extraction of the corresponding boolean function cannot be made manually. A program to automatically generate the diagnosis equation was implemented and the result has 300 terms. The minimal equation was generated by the SIS [9] and has 228 product terms. Again, it is not possible to choose manually one minimal term. All terms must be compared in order to find the one with the fewest literals and providing maximal fault coverage. Thus, a program to determine the minimal terms and the one to be chosen should be developed.

7.2 Diagnosis by Phase Measurement

Similarly to section 7.1, a new table considering interconnect faults and the new ranges is built for the phase parameter (using tables 1,2 and 4). The dimensions of the resulting matrix are 31x24: 31 detection ranges and 24 non-equivalent faults. The corresponding equation has 299 terms, but the minimal expression generated by SIS has 48 terms, from which 32 minimal terms with the same number of literals (18). The same conclusions made above apply here.

7.3 Diagnosis by Gain and Phase

This is the most complex approach since all detection ranges are considered together and are rearranged in order to isolate each fault considered. But it is not possible to consider simultaneously the ranges for which gain is measured and the ranges where the phase is measured. Thus, the resulting table has a list of ranges A_i (gain) followed by a list of ranges B_i (phase) and the minimal set generated is of type A_iB_i, which means that both parameters must be observed at the output. The generated table has 58 ranges and 30 non-equivalent faults and the corresponding equation has 435 terms. This equation is used as input to the SIS tool running in a Sparc Station 1+ with 28M of RAM and system Sun4. The system worked for 4 days and, in this period, did not and a minimal equation.

The results above show the complexity involved in the treatment of interconnect faults in diagnosis. The algorithm to generate the boolean function from the tables of detection ranges per fault is $O\ (n^2 \ .\ k)$, where n is the number of faults and k is the number of detection ranges. This shows that the ratio between the number of ranges and faults and the processing time is not linear, and the problem becomes complex even for simple circuits as a biquad filter. Considering boards and commercial designs, the number of interconnect faults will explode. Also, the dimensions of the equations generated and the time needed to minimize them show that it is necessary to implement an heuristic to perform the simplification of the boolean equation.

The development of such a heuristic is ongoing; a first solution is to split the diagnosis process into two steps: the first one will show if the fault is in a component or if it is in an interconnect. This can be done by simply adding the final detection range for the interconnect faults ([1.48k..5k]) in the table of the phase parameter. Using this approach, the final diagnosis ranges are capable of diagnosing any component faults (indicating which component) and interconnect faults. If the fault is an interconnect one, the second step takes place: new diagnosis tables, for gain and phase parameters, with the detection ranges only for interconnect faults are built and analyzed to lead to the actual location of a short.

8 Computer-aided testing tools

The test methodology proposed in sections 3 and 4 is based on the possibility of optimizing and automating the test process in order to lower testing costs. Some commercial tools used in this work for symbolic computation (Maple V) and for electrical simulation (HSpice) were mentioned in previous sections.

For the transfer function definition, tools such as SSpice and XFUNC can be used, both running in a DOS environment. They receive a structural Spice-like description of the circuit and generate the corresponding transfer function.

The sensitivity computation is made over the transfer function using some mathematical definitions. This way, any tool capable of implementing differential calculus and of plotting features and curves can be used to improve the test generation process. This work used Maple V, a powerful tool available in-house that runs in UNIX, although there are commercial versions for WINDOWS. MATHCAD is another possibility that can be used in a WINDOWS environment.

The tools briefly described above can generate the first detection ranges that will be used to determine the coverage of interaction faults. For this analysis, an electrical simulator is needed. HSpice, a standard simulator running in UNIX was used, since the first structural description was made in its input language. This tool is also available in DOS.

In order to fully automate the test process, some additional tools are also needed. They will interface the commercial tools in use by way of parsers. Besides that, a program that determines the coverage of interaction faults and defines new ranges to undetected faults is essential and is now under development.

For the diagnosis process, the SIS tool can be used to minimize the equations extracted from the diagnosis tables. Although this tool can perform other tasks, such as synthesis, the feature used in this application is the minimization procedure. Any other tool that can minimize boolean functions can be used here. However, due to the complexity of the diagnosis equations, a heuristc is needed for defining and simplifying these equations before the minimization step is applied.

9 Concluding remarks

This paper has presented a test generation methodology for analog circuits. A sensitivity-based test generation tool was proposed that can automatically generate a minimal set of test signals that guarantees maximum fault detection. The shorter the test set, the cheaper the test process, since less test stimuli must be applied to the circuit under test and less test parameters must be measured. The fault model used considers interaction shorts in addition to soft and large component deviations, and hard faults (opens and shorts).

The test methodology was illustrated by means of a case-study, a biquadratic low-pass filter. The test parameters considered were the gain and the phase of the filter. The gain showed to be the best parameter to detect faults in components and, the phase to be the best to detect interaction shorts. The method was also applied to a non-linear circuit, a voltage controlled oscillator (555-like oscillator) [5]. In this case, the test parameter measured was the output frequency. Two values for the voltage control (2.5V and tri-state) were enough to detect every fault in the model.

This work has also presented an approach to achieve diagnosis of component and interconnect faults based on an initial set of test vectors. It proved to be very complex, even for a simple circuit as the biquad filter, due to the consideration of interconnect faults.An heuristic to solve this problem, based on splitting the diagnosis procedure into two steps, is being studied and tested in other circuits in order to validate its results.

The commercial tools presented in the last section show that it is possible to automate the test procedure and build a CAT tool capable of generating a minimal test set. This test set is able to detect both, component and interconnect faults and, at least, determine which kind of fault is present in the circuit. For component faults the tool can completely define which component is faulty.

The test generation process, including the diagnosis procedure, shall be fully automated in a vey short term.

References

[1] B. Ayari, N. Ben Hamida and B. Kaminska. Automatic Test Vector Generation for Mixed-Signal Circuits. *Proc. European Design and Test Conference* , 458-463, 1995.

[2] N. Ben Hamida, B. Kaminska. Analog Circuit Testing Based on Sensitivity Computation and New Circuit Modeling. *Proc. IEEE International Test Conference*, October 1993.

[3] Y. Lu, R. Dandapani. Hard Faults Diagnosis in Analog Circuits Using Sensitivity Analysis. *Proc. IEEE VLSI Test Symposium*, 225-229, 1993.

[4] P.Caunegre and C.Abraham. Fault Simulation for Mixed-Signal Systems. *Journal of Electronic*

Testing: Theory and Applications, 8: 143 - 152, 1996.

[5] E. F. Cota and M. Lubaszewski. ATPG para Teste de Circuitos Analógicos e Mistos. CPGCC da UFRGS, march 1997. (Masters Thesis)

[6] B. W. Char et al. *Maple V - Language Reference Manual*. Waterloo Maple Publishing, Waterloo, CA, 1994.

[7] S. Mir, M. Lubaszewski and B. Courtois. Fault-Based ATPG for Linear Analog Circuits with Minimal Size Multifrequency Test Sets. *Journal of Electronic Testing: Theory and Applications*, 9: 43 - 57, 1996.

[8] N. Nagi, A. Chaterjee, A. Balivada and J. A. Abraham. Fault-Based Automatic Test Generator for Linear Analog Circuits. *Proc. ACM/IEEE International Conference on Computer Aided Design*, 88 - 91, 1993.

[9] Sentovich et al. SIS: A System for Sequential Circuit Synthesis. (Memorandum no. UCB/ERL M92/41), Electronics Research Laboratory, Department of Electrical Engineering and Computer Science, University of California, 1992.

[10] Slamani and B. Kaminska. Multifrequency Analysis of Faults in Analog Circuits. *IEEE Design & Test of Computers*, 12(2): 70 - 80, 1995.

[11] R. W. Liu (Editor). *Testing and Diagnosis of Analog Circuits and Systems.* Van Nostrand Reinhold, New York, 1991.

[12] N. O. Nagi and J. A. Abraham. Hierarchical Fault modeling for Analog and Mixed-Signal Circuits. *Proc. IEEE VLSI Test Symposium*, 96-101, 1992.

[13] N. Nagi, A. Chatterjee and J. A. Abraham. DRAFTS: Discretized Analog Circuit Fault Simulator. *Proc. ACM/IEEE Design Automation Conference,* 505-514, 1993.

Sociedade Brasileira de Computação

Sociedade Brasileira de Computação - UFRGS
Av. Bento Gonçalves 9500, B. Agronomia
Caixa Postal 15064
91501-970 Porto Alegre, RS - Brazil
Tel. / Fax: (55 51) 316.6835

jbcs@icmc.sc.usp.br