Minimal Length Diagnostic Tests for Analog Circuits using Test History

Alfred V. Gomes and Abhijit Chatterjee
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
{vgomes,chat}@ee.gatech.edu

Abstract

In this paper we propose an efficient transient test generation method to comprehensively test analog circuits using minimum test time. A divide and conquer strategy is formulated to sequentially synthesize the test stimulus for the entire duration of test. We use a novel measurement procedure to resolve ambiguities in the present measurement sample by using class association information from the previous samples. This sequential formulation of test generation problem enables fault dropping and greatly reduces simulation and optimization effort. Additionally, this method is immune to noise and tests can be easily calibrated for use in hardware testers.

1 Introduction

The rapid evolution of IC technology has placed increased demands on issues relating to efficient techniques to test and diagnose mixed signal integrated circuits. Typically these circuits have complex specifications, as small device geometries permit integrating multiple functions in a single device. Production tests which are based on explicit verification of these specifications are expensive and time consuming. In recent years, researchers have tried to emulate the success of test techniques for digital circuits in the analog domain. Although the goals are similar, test techniques for digital circuits cannot be directly applied to analog circuits. Test generation and diagnosis for analog circuits is viewed as a tougher problem with many issues still unresolved.

2 Problem statement and key contributions

Test and diagnosis of analog circuits has gained increased importance due to difficulties involved in testing embedded analog blocks in larger digital circuits. The main problems are a) limited accessibility to internal nodes b) isolation of analog and digital blocks may not be possible and c) embedded analog blocks may not have distinct self-contained specifications, hence fault models and pass/fail criteria has to be developed if these blocks are to be separately tested. In this paper we propose a test stimulus generation and response measurement method which is used to detect and identify faults in analog blocks with a) Minimal length transient test sequences b) Generate input stimuli with simple pulse shapes which can be easily applied in a mixed signal environment c) A novel response encoding method which uses test history to identify circuit failures d) Tests can be developed for GO/NO-GO production tests or to obtain diagnostic information and e) Fault dropping is used to reduce simulation and optimization complexity.

This work has been supported by DRAPA under contract F33615-95-2-5562

3 Previous work

Initial research work in this field concentrated on efficient application of specification tests by determining the optimal sequence[1] of tests which minimizes average test time and dropping redundant tests by exploiting high correlations between tests[2]. Inexpensive *alternate tests* were proposed to partially test complex circuits. These tests do not directly test the functionality or the specifications of the device, but is necessary (but may not be sufficient) to pass the test if functional and performance specifications are to be met. Inexpensive DC based alternate tests are proposed in [3-4]. These tests, used at wafer-probe stage can quickly isolate defective devices. DC tests are limited by their inability to exercise all operating modes of a circuit. Alternate tests based on AC stimulus are proposed in [5-9]. AC tests, in general are used when circuits have a dominant signal flow characteristic (e.g. filters) and specifications are strongly related to the frequency domain behavior of the circuit.

Transients tests[10-14] specify a time domain waveform (stimulus), which may be periodic and a detection criteria for the entire duration of the test. For linear circuits, [10] uses a quadratic programming formulation to maximize the difference in response between good and faulty circuits, while [12] computes hyperplanes in the measurement space to discriminate between good and faulty circuits. Both the methods require impulse response of the system to be known. [13] uses saturated ramp as the test stimulus for linear circuits. A minimax formulation is used in [11] to maximize the difference between a good and a faulty circuit using the integral of the weighted response. Since the entire time domain response is quantified by a single quantity, the test resolution will be poor. The transient test generation methods in [10,12,14], strongly depend on the linear properties of a circuit, and hence are restricted to circuits with linear behavior.

Although transient tests for analog circuits are computationally difficult to develop, they are a supersets of DC &

AC tests, and hence have the capability to detect larger number of faults. In this paper we propose a new transient test generation method for general analog circuits. This formulation permits sequential test generation, fault dropping and high measurement resolution. The rest of the paper is organized as follows: In the next section, we present the fault model and the test generation methodology. Implementation and experimental results are described in sections 5 and 6. Section 7 concludes the paper.

4 Proposed test generation method

The operation of test generation program proposed in this paper is divided into two phases. a) Construction of statistical fault model and b) Computation of input stimulus and response measurement.

4.1 Fault model description and construction

Alternate test generation methods require a fault model to be defined. A fault model permits abstraction of the performance specifications of Circuit Under Test (CUT), as admissible range of values, which the components (or nodes) can assume without violating the specifications. Since a specification in an analog circuit may depend on multiple component values, a faulty range for a component is dependent on the values of other components. To capture component interdependencies, a statistical fault model uses multi-variate distributions of the component parameters[12,15]. These distributions are generated by using process defect information. In figure 1, the parameter space represents the circuit instances characterized by their component parameter values and the dimension of this space depends on number of variables considered. To identify faulty circuits in the parameter space, all circuits are mapped to the specification space by simulation of the circuit specifications[16]. Since an acceptability region in the specification space is known, we classify circuits in the parameter space as faulty if it maps outside this region. This fault model conveniently describes faults in terms of circuit parameters, while retaining a strong relation to specifications.

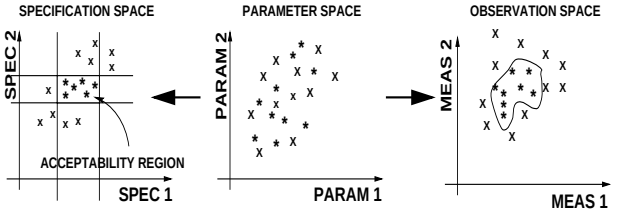

FIGURE 1. Statistical fault model

This stage of test generation produces a set of circuits belonging to faulty and fault-free classes. The circuits in the fault free class accounts for the tolerance effects in analog circuit. The conventional fault-based test generation method omits this critical step and directly uses a fault list which maybe derived by injecting large changes in component values. As these faults have no direct relation to specifications, they may or may not cause a specification violation. Fault-based testing can be accommodated in the our method by adding the fault list to the list of circuits in the parameter space of the statistical model before mapping to specification space.

Given a set of circuits, classified as faulty and fault-free (obtained from the fault model), the test generation problem can be posed as: *computation of an input stimulus such that, a measurement criterion can be defined which will correctly discriminate the two classes of circuits in the observation space (figure 1), within the constraints imposed by the test environment.* This mapping depends on two user controlled parameters: input stimulus and the set of observations, thus providing two degrees of freedom for test generation.

If the above mapping is not accurate, there will be misclassifications in the observation space. *These misclassifications are quantified by fault and yield coverage estimates where fault(yield) coverage is defined as the probability that a faulty(good) circuit fails(passes) the alternate test.* All test generators attempt to maximize fault and yield coverage.

It is important that the sample population chosen should represent various fault effects and failure modes accurately. It may be noted that if the distribution of the training set used, is different from the distribution of defects that actually occur, but still captures the effects of process level variations, the alternate tests are still valid, but the fault and yield coverage *estimates* may be inaccurate. Sampling techniques and statistical tests on distributions should be used to validate the size and representation of the training set.

4.2 Computation of input stimulus and response measurement

The test generation problem as posed in the previous section requires a search for a combination of input stimuli and a measurement criterion among all admissible combinations and this is clearly infeasible for any non-trivial circuit. To make this problem tractable, we pre-define the measurement procedure and then optimize the input waveform to obtain maximum fault and yield coverage. This still remains a difficult problem as the input is a time domain waveform and has to be chosen among all possible waveforms (an infinite set).

Input stimulus generation involves obtaining a waveform for the duration of the test, within the constraints imposed on the waveform. Rather than optimizing the waveform for the entire duration of the test, we use a *divide and conquer* strategy, whereby the input is sequentially synthesized by

considering a small time interval of the total test time. Let this interval be T. The choice of T is related to the time required to induce significant activity in the nominal CUT. For e.g, for filters, T is chosen equal to the largest time constant while for ADC it maybe set equal to the conversion time.

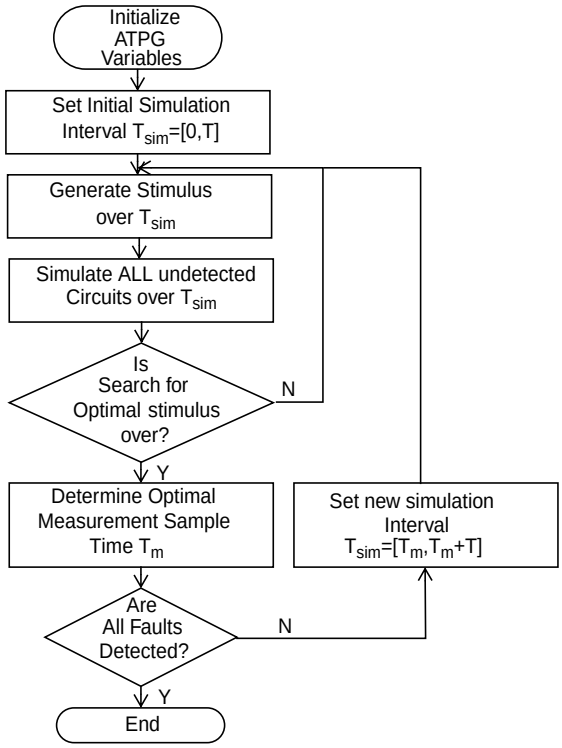

FIGURE 2. Flow chart of the main routine

The flow chart of the main top level test generation procedure is shown in Figure 2. This procedure contains two major subroutines, the stimulus generation subroutine and the measurement subroutine, which are described in detail in the following sections. Consider the initial interval, T_{sim} =[0,T] for which the test stimulus is to be generated. The stimulus generation routine selects a prospective input stimulus and all undetected circuits are simulated over T_{sim}. The measurement routine is next invoked to analyze the response over T_{sim} and determine the first best sample point T_m, which will correctly classify maximum number of circuits. The circuit state at T_m is also saved to restart the simulation if needed. The main loop iterates using different input stimuli over T_{sim} till finding an input stimulus which correctly identifies all circuits or all input stimuli are exhausted (the search is restricted to a finite set of pulse shapes). The stimulus which identifies the maximum number of circuits is chosen and sample point T_m is noted. All circuits which are correctly identified in this interval are dropped and the simulation interval is incremented to T_{sim}=[T_m,T_m+T]. Using the saved circuit final state from the previous interval, next simulation is restricted to T_{sim} only.

This structure of the test generator gives the flexibility to define external routines for stimulus generation and measurement, which is critical in analog ATPG as an analog circuit may have specific stimulus and measurement demands.

4.2.1 Input stimuli generation

Consider a simulation interval T_{sim}=[T_1,T_2] for which the test stimulus is to be generated. An undirected search for such a functional over the entire input space is not computationally feasible. Rather than search for an arbitrary shaped function, we consider a set of easily parameterizable activation functions(e.g step, exponential, ramp, sinusoid etc.) and compute the parameters of these functions which will maximize the number of faults that are detected.

For the application in this paper we use the step waveform as the activation function. Step waveform is chosen because of its high spectral content and hence its ability to excite a large range of faulty conditions, and additionally, generation of the waveform is easy. The maximum and the minimum amplitude of the waveform is fixed by the allowable range of the input signal.

Assume that the initial voltage at time T_1 be V_{max}. We need to determine a transition time point T_p in T_{sim}, at which the state of the waveform has to be switched from V_{max} to V_{min} (or from V_{min} to V_{max} if initially the waveform is at V_{min}.) which would result in correctly identifying maximum number of circuits by the measurement procedure. This formulation reduces to optimization of single continuous variable T_p. Variation of the cost function (evaluated by the measurement routine) with respect to T_p is “smooth” (although integer valued) but may not be monotonic. To avoid converging to a local solution, an initial set of transition points T_p, spaced equally on the interval T_{sim}, is chosen. If we generate n-1 points, T_p is the set {(T_{sim}/n)*i, i=1 to n-1}. The response for each waveform defined by this set is evaluated, and the waveform for which the cost function is maximum is selected. Suppose this point is T^*_1, we assume that the probability of finding the maxima over the entire interval is highest in the neighborhood of T^*_1 and varies monotonically. This heuristic is valid if “sufficient” number of points are chosen in the initial search. A modified binary search is carried out by generating two stimulus with transition points at T^*_1+T/2n and at T^*_1-T/2n, and choosing the waveform which maximizes the cost function. Let this point be denoted as T^*_2.

In the next stage, we choose T^*_2+T/4n and T^*_2-T/4n as the transition points, iteratively refining the interval. In our implementation the binary search is terminated after traversing 4 levels, as the increase in cost function is not significant compared to cost of search.

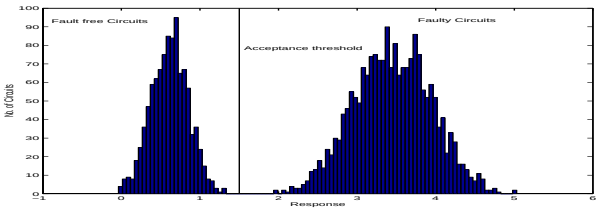

FIGURE 3. Ideal distributions of faulty and fault free responses

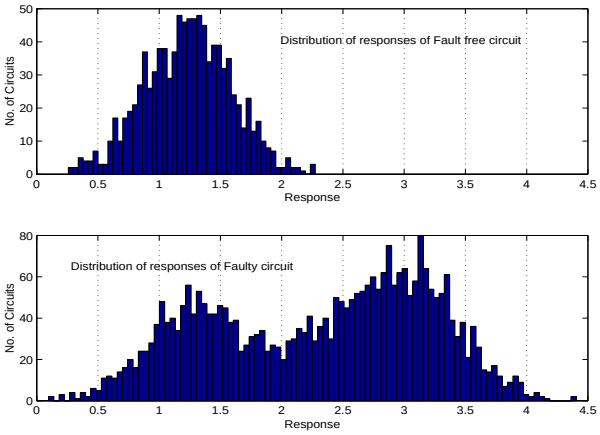

FIGURE 4. Distributions of fault and fault free responses, which overlap

4.2.2 Measurement procedure

For a simulation interval T_{sim}, the response of all circuits simulated in this interval can be viewed a distribution, which is a function of the underlying component distributions, the circuit function and the input stimulus. In the ideal case we would like the distribution of the response of the faulty and fault free circuits to be distinct (at a particular sample point), so that a simple threshold can be used to classify the circuits (figure 3). But, in reality, the distributions may overlap, and if an observation of the response lies in this region, it cannot be correctly classified based on the information from that sample alone. Figure 4 shows a distribution of the *output responses at a particular sample point* for the biquad filter, which we use to demonstrate our technique in section 6.

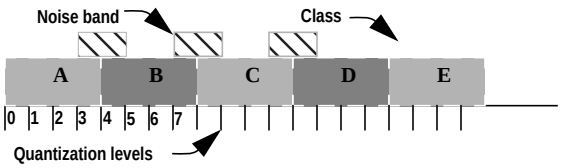

FIGURE 5. Grouping of quantization levels into classes

The simulator output, for a given simulation interval T_{sim} is sampled and quantized for all circuits which were simulated. Measurement procedure uses test history and class association information to decide on a sample point which will identify maximum number of circuits as described below. For every sample point in T_{sim}, the quantized response of circuit is assigned to a particular class, where classes are sets of consecutive quantization levels grouped together (figure 5). For e.g, if a sample is quantized into 4096 levels using a 12 bit ADC, then sets of 16 or 32 quantization levels can be grouped together to obtain 256 or 128 classes respectively. Responses of circuits for a candidate stimuli, at a given time point, are assigned to a class depending on magnitude of quantized response. After mapping, a particular class may contain only fault free circuits, or only faulty circuits, or a mix of faulty and fault free circuits or the class may be empty. For circuits which map to a class containing faulty and fault free circuits, *test history* of the circuit in the previous samples is used to determine if a particular circuit has a unique signature. For e.g, if in the *i* th. sample (see figure 6), circuit A which is fault free, and circuit B which is faulty, map to a class *k (hence cannot be identified)*, and in the *i-1* th. sample, A maps to *m* and B maps to *n* (classes *m* and *n* contain circuits of mixed type, otherwise A and B would have been dropped in the *i-1* th. sample), A and B can be correctly classified based on class information in the prior samples. Test history is implemented as an association list for each for each circuit. At the initial time point, the association list for each circuit will contain all other circuits (the circuit is indistinguishable from other circuits). A particular circuit is identified as faulty or fault free if the intersection between elements in the association list and the elements in the class in which the circuit belongs results in a set having elements of same type, then the circuit is determined to be correctly identified. If that sample point is chosen, the resultant set from the intersection is used as the new association list. As new samples are added, the number of entries in the association list drops. Effects and handling of measurement noise is described in section 5. If the test generator is used to generate diagnostic tests, the sample sequence is accumulated till the association list contains only one element, hence each fault will have a unique trace sequence.

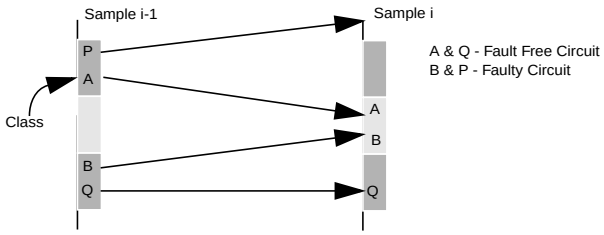

FIGURE 6. Correct identification circuits A and B due to unique response sequence

4.2.3 Detection trace sequences

From the previous section, a time domain waveform, specified by the transition points is obtained, along with a time vector which specifies the time at which the response is to be sampled. To use this input waveform, we need to specify a classification criteria, to accept or reject a circuit. This criteria is specified as a trace sequence, where specifies a set of sequence of classes, which terminates either in pass, fail or unresolved states. These sequences are extracted and stored by the measurement procedure, when it attempts to resolve a circuit either as faulty or fault free. The sequences which are unresolved at termination of test generator are labeled as unresolved. These unresolved states exist, either because these circuits map very close to the boundaries of the specification tests or the test stimulus and measurements in the alternate test are not sensitive to certain behavior of the circuit. Detection trace sequence can be implemented in the test equipment in either software sequence detectors or programmed into a FPGA as a state machine.

5 Implementation details

The implementation of the test generator uses a mix of commercial and custom built tools. We use Spectre[17] as the core circuit simulator. SKILL and Matlab functions are used for data analysis. A C based test generation control routine is used to control and synchronize the different tools. Since the computational complexity of the test generator is due to the high cost of circuit simulation, an efficient and powerful simulator is essential. Spectre supports saving of state files, which permits stopping and restarting a transient simulation. When we search for the optimal input in a interval, Spectre makes use of the state information saved form the previous interval, and simulates for the duration of the interval only. Additionally Spectre supports behavioral modeling of circuits, which permits test generation for large circuits.

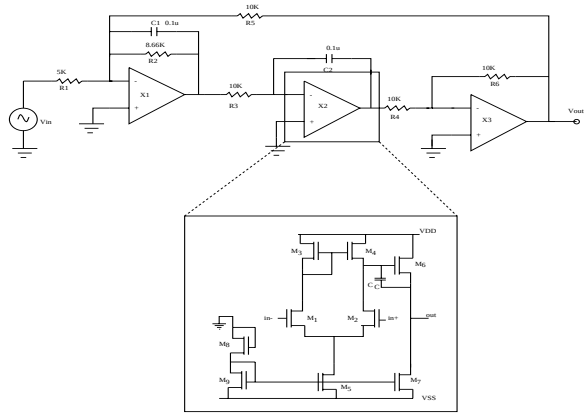

FIGURE 7. Schematic of low pass biquad filter

In practical test environment, presence of measurement errors, may lead to misclassification of responses. We assume that systematic errors in measurement have been removed by calibration, and the resultant measurement noise is zero mean and normally distributed. The RMS value of measurement noise is be assumed to be comparable to the resolution of the measurement equipment (otherwise using a high resolution instrument will not serve any purpose). It is seen that, measurement noise affects only those responses which map near the boundary of two classes. Since this uncertainty prevails only in a few quantization levels around a class boundary, a set of sub-classes are defined around these boundaries (figure 6). A circuit which maps into a noise sub-class will be associated with both the neighboring classes, but that circuit will not be evaluated for detectability at that sample. Large number of classes in presence of significant noise will lead to poor performance. The width of noise sub-classes is dependent on actual noise in the test set-up.

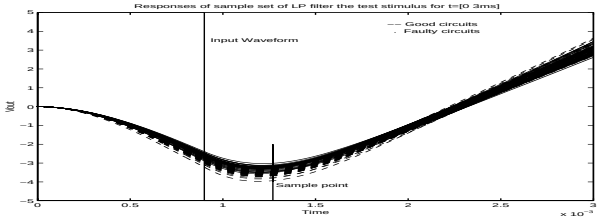

FIGURE 8. Distribution of the responses for the interval [0, 3ms

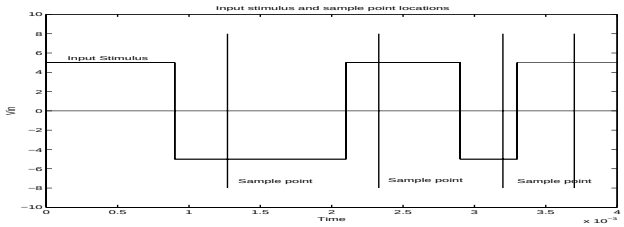

FIGURE 9. Test wave form computed by the alternate test

6 Experimental results

In this section, we present the fault simulation results for the biquad low pass filter shown in figure 7. This filter uses three identical CMOS opamps. The filter specifications are listed in table 1

Simultaneous variation in multiple parameters, shown in table 2, is considered, where each of the parameters is assumed to be normally distributed with a 3σ variation of 10%. A training set of 200 circuits is generated from this parameter space. For each of these circuits, a full circuit simulation is performed to determine the filter specifications and depending on the conformance to specifications, the circuits are classified as fault free or faulty. This training set is used develop an alternate test criteria as proposed in this paper. Figure 8, shows the response of the training

set to the test stimulus in the first interval and location of the sample point (vertical line). Table 3, lists the set of parameters used by test generator. Figure 9, shows the test waveform for the entire duration of the test, with location of the sample points marked by vertical lines. Table 4 list the achieved fault and yield coverage.

TABLE 1. Low pass filter specification

Name	Specification
DC Gain	2 +/- 10%
3dB bandwidth	190Hz +/- 10%
Maximum gain	< 2.2

TABLE 2. Circuit parameters used in the fault model

Circuit Parameters	Transistor parameters
R1, R2, R3, R4, R5, R6, C1, C2	V_t, t_{ox}

TABLE 3. Test Generator parameters used for the filter

Parameter	Value
Interval T	3 ms
Quantization levels	4096
Classes	256
Vmin	-5V
Vmax	5V
Cardinality of t*	10
Binary search level(i)	4

TABLE 4. Fault and yield coverage

Number of circuits	Fault Coverage	Yield Coverage
200	0.99	0.995

7 Conclusion

This paper presents a systematic procedure to develop time domain tests which can comprehensively test large analog circuits. We propose some reasonable constraints on the input stimulus and develop a powerful measurement criteria to make time domain test generation computationally tractable for general circuits. The test generator, in addition to classifying the circuits into good and fault free categories, can also provide information on type of failures that actually occur by using the information from the trace sequences. This is useful in diagnosing dominant failure modes in a process. We believe this method will provide the base framework to develop tests for multiple input and mixed signal circuits. Interactions between different inputs and between the digital and analog portions of a circuit can analyzed in a small time interval, so that faults can be excited and fault effects propagated to the primary outputs.

References

[1] L. Milor and A. S. Vincentelli, "Minimizing production test time to detect faults in analog circuits," *IEEE Transactions on Computer Aided Design*, pp. 796-813, Vol. 13,1994.

[2]. B. Brockman and S. W. Director, "Predictive subset testing: optimizing IC parametric performance for quality, cost and yield," *IEEE Transactions on Semiconductor Manufacturing*, pp. 104-113, 2, 1989.

[3] M. J. Marlett and J. A. Abraham. "DC-IATP: An iterative analog circuit test generation program for generating DC single pattern tests," *International Test Conference*, pp. 839-845, 1988.

[4]L. Milor and V. Viswanathan, "Detection of catastrophic faults in analog integrated circuits," *IEEE Transactions on Computer Aided Design*, pp. 114-130, Vol. 8,1989.

[5]N. B. Hamida and B. Kaminska, "Multiple fault analog circuit testing by sensitivity analysis," *Journal of Electronic Testing: Theory and Applications*, pp. 331-343, 1993

[6] W. Mao,Y. Lu, R. K. Gulati, R. Dandapani and D. K. Goel, "Test generation for linear analog circuits," *IEEE 1995 Custom Integrated Circuits Conference*, pp. 521-524, 1995.

[7]M. Salamani, B. Kaminska and G. Quesnel, "An integrated approach for analog circuit testing with minimum number of detected parameters," *International Test Conference*, pp. 631 - 640, 1994.

[8]M. Salamani and B. Kaminska, "Multifrequency analysis of faults in analog circuits," *IEEE Design & Test of Computers*, pp. 70-80, Summer 1995.

[9]A. Abderrahman, E. Cerny and B. Kaminska, "Optimization based multifrequency test generation for analog circuits," *Journal of Electronic Testing*, pp. 59-73, Vol. 9, 1996.

[10]S. J. Tsai, "Test vector generation for linear analog devices," *International Test Conference*, pp. 592-597, 1991.

[11]G. Devarayanadurg and M. Soma, "Dynamic test signal design for analog ICs," *International Conference on Computer Aided Design*, pp. 627-630, 1995.

[12] C. Y. Pan and K. T. cheng, "Implicit functional testing for analog Circuits," *VLSI Test Symposium*, pp. 489-494, 1996.

[13] A. Balivada, J. Chen and J. A. Abraham. "Analog testing with time response parameters," *IEEE Design and Test of Computers*, pp. 18-25, Vol 13, 1996.

[14]Pramodchandran N. Variyam and Abhijit Chatterjee, "Test generation for comprehensive testing of linear analog circuits using transient response sampling," *International Conference on Computer Aided Design*, pp. 382-385, 1997.

[15]W. M. Lindermeir, "Design of robust test criteria in analog testing," *International Conference on Computer Aided Design*, pp. 604-611, 1996

[16]A. V. Gomes, R. Voorakaranam and A. Chatterjee, "Modular fault simulation of mixed signal circuits with fault ranking by severity," *IEEE International Sym. on Defect and Fault Tolerance in VLSI systems*, pp. 341-348, 1998.

[17]Cadence, "Spectre User Guide", CDS Inc. 1995