Prediction of Analog Performance Parameters Using Fast Transient Testing

Pramodchandran N. Variyam, *Member, IEEE*, Sasikumar Cherubal, *Member, IEEE*, and Abhijit Chatterjee, *Senior Member, IEEE*

***Abstract*—In this paper, a fast transient testing methodology for predicting the performance parameters of analog circuits is presented. A transient test signal is applied to the circuit under (cut) test and the transient response of the circuit is sampled and analyzed to predict the circuit's performance parameters. An algorithm for generating the optimum transient test signal is presented. The methodology is demonstrated in a production environment using a low-power opamp. Result from production test data showed: 1) a ten times speedup in production testing; 2) accurate prediction of the performance parameters; and (3) a simpler test configuration.**

***Index Terms*—Analog specification prediction, mixed-signal circuit testing, test generation.**

I. Introduction

TRADITIONALLY, the quality of an analog IC is evaluated by measuring its *performance parameters*. Slew rate, offset, gain bandwidth, etc. are examples of performance parameters of an opamp. The measured performance parameters are compared with the *specification* limits to decide if an IC is good or bad. A lower bound on the slew rate, upper bound on offset, etc., are examples of the specifications of an opamp. This testing methodology is referred to as *functional testing or specification-based* testing. During production, ICs are first subjected to wafer probing, wherein a set of low-frequency functional tests is performed. Good ICs are then packaged and subjected to a final test, comprised of a complete set of functional tests. A sample set of qualified packaged ICs is subjected to stress testing by operating the ICs in an oven at elevated temperatures for a long time. This is to eliminate any "weak" devices that are likely to fail during their early period of operation. Finally, a smaller sample of packaged ICs is retested prior to delivery for quality assurance (QA) purposes. Thus, functional tests are performed repeatedly during the wafer probe, final test, stress test, and quality assurance test. This repeated performance of costly functional tests increases the production test cost of analog and mixed-signal products [1]–[4]. This paper presents a low-cost testing methodology that can replace costly functional testing. These low-cost tests are referred to as *alternate tests* in this paper. Experiments performed on a low-power operational amplifier test vehicle in a production environment showed that up to an order of magnitude reduction in production test time can be achieved with the proposed testing methodology. In the recent past, there has been a lot of research work in low-cost alternate testing of analog and mixed-signal circuits. Below, a brief overview of the previous research is given.

A. Previous Research

In [5]–[8], researchers proposed that the performance parameters have a high correlation between them and hence many functional tests can be eliminated from the test plan without affecting the overall test quality. During production testing, not only the number of functional tests, but also the order in which they are performed, affect the overall testing time. Hence, in [9] and [10], a further reduction in the average production testing time is achieved by appropriately ordering the functional tests. Although these techniques [5]–[10] helped to reduce the production testing time by eliminating many functional tests and by ordering them in an appropriate manner, they used costly functional tests for fault detection.

Motivated by the popularity of *fault-based testing* in the production testing of digital circuits, many researchers tried replacing the functional tests with simple fault-based alternate tests in the analog and mixed-signal domain. In fault-based testing, a list of physically realistic faults are derived from process information, defect statistics, and circuit layout [11]. Tests are then developed to distinguish these faulty circuits from the fault-free circuit. Depending on the nature of the test signal and the way in which the circuit response is analyzed, fault-based testing can be classified into three categories.

1) Static DC Testing: In the dc testing, a dc voltage or current is applied to the CUT and the dc response of the circuit is monitored to detect faults in the CUT [12]–[15]. Although dc tests can detect catastrophic faults effectively, they cannot detect parametric failures effectively.

2) Steady State Frequency Domain Testing: In the frequency domain testing, a sinusoidal signal of a certain frequency is applied to the CUT and the steady state response of the CUT is monitored for fault detection [16]–[24]. Although steady state ac tests can detect parametric faults effectively, applying many sinusoidal signals and studying the steady state response of the CUTs is time consuming.

3) Time Domain Transient Testing: In the time domain testing, a transient signal like a multitone signal or a piece-wise linear (PWL) is applied to the CUT using an arbitrary waveform

Manuscript received August 23, 2000; revised February 22, 2001. This paper was recommended by Associate Editor C.-K. Cheng. This work was supported in part by National Semiconductor Corporation under a research grant and by the U.S. Defense Advanced Research Project Agency (DARPA) under Contract F33615-95-2-5562.

P. N. Variyam is with Texas Instruments Incorporated, Dallas, TX 75243 USA.

S. Cherubal is with Ardext Technologies, Atlanta, GA 30332 USA.

A. Chatterjee is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA.

Publisher Item Identifier S 0278-0070(02)01783-9.

0278-0070/02$17.00 © 2002 IEEE

generator (AWG) and the transient response of the CUT is sampled and analyzed for fault detection [25]–[30].

In the fault-based test generation methods described above [12]–[30], test generation problem is formulated as an optimization problem where the goal is to maximize the difference between fault-free and faulty circuit responses. The optimization problem is solved in a computer-aided framework to determine the best input stimulus. The CUT is excited with this input stimulus and its response is compared with a threshold for detecting faults. The chief drawback of these methods is that they do not correlate the alternate test measurements with the performance parameters. Hence it is not possible to compare and contrast the quality of these alternate tests with that of the functional tests.

In an attempt to overcome this drawback, many researchers linked the alternate test thresholds directly with the circuit specifications such that if a circuit passes the alternate tests it is guaranteed that it meets all the specifications. In [1] and [31], a set of dc and low-frequency test measurements were used to check for circuit specifications. In [32], white noise was used as an alternate test signal to test linear time invariant (LTI) analog circuits. In [33], a test generation problem was formulated as a problem of deriving hyperplanes and the coefficients of the derived hyperplane were used as test patterns for LTI circuits. In [34], sensitivity-based approximations were used to derive alternate tests for linear analog circuits. In all the methods described above [1], [31]–[34], test decisions were made by looking at a linear combination of the test measurements. A linear combination of m test measurements define a hyperplane in an m dimensional space. These hyperplanes are derived such that all the circuits satisfying a particular specification is on one side of the hyperplane and all the circuits that violate the specification is on the other side of the hyperplane in this space. A CUT is declared as faulty or fault-free depending on which side of the hyperplane the alternate test measurements lie.

B. Contributions of the Proposed Approach

Although the testing methods in [1] and [31]–[34] can check if a circuit satisfies the specifications or not, these tests cannot be used to get any information about the performance parameters of the circuit. The test engineer has to decide about the quality of the product by looking at an abstract quantity such as a linear combination of the test measurements [1], [31]–[34]. In the proposed testing method, the *performance parameters of the CUT is predicted from the transient response of the CUT*. This is of great advantage during production as the performance parameter values provide physically interpretable information about the quality of the product. This helps in making a confident decision about the quality of the CUT by comparing the calculated performance parameters against the specification bounds. Moreover, all the performance parameters of the CUT are calculated from fast transient tests leading to a considerable reduction in production testing time. The testing methodology presented here is applicable to general nonlinear analog circuits and mixed-signal circuits. The proposed testing method consists of the following steps.

1) A PWL transient test stimulus is generated in a computer-aided framework such that the chances of misclassifying a circuit during testing are minimized. Many researchers have used PWL test stimulus for detecting faults in analog circuits [28]–[30], [34]. However, the objective in all these methods was to maximize the difference between the response of the fault-free and faulty circuits in a fault list. The objective of the testing technique presented here is to detect circuits that fail the specifications without actually performing the specification tests. In such a scenario, circuits with performance parameters close to specification boundary are most prone to get misclassified. Hence, maximizing the difference between the response of the fault-free and faulty circuits in a arbitrary fault list does not help. Hence, the test generator presented in this paper specifically targets circuits close to the specification limits.
2) The mathematical relationships between the transient response of the circuit and the performance parameters are derived using statistical techniques and stored in the tester computer.
3) The alternate transient test is performed on the CUT and the performance parameters of the CUT are predicted from the transient test measurements using the relationships stored in the tester computer.
4) A pass/fail decision is made based on the predicted performance parameters of the CUT.

The test generation algorithm given in this paper can generate PWL test waveforms to excite only one input of the CUT. To accurately predict the performance parameters of an IC it might be necessary to excite more than one input of the IC using PWL transient waveforms. For example, to predict the common mode rejection ratio (CMRR) of an opamp, it is necessary to excite both the positive and negative input terminals of the opamp. If a PWL waveform is applied at one terminal with the other terminal at a fixed potential, the transient response will not contain the information necessary to predict the CMRR. However, generating PWL test stimulus for all the inputs of a CUT can be computationally expensive. Hence, such performance parameters are predicted by exciting the IC using the PWL stimuli generated using designer input.

Five performance parameters of a low-power operational amplifier from National Semiconductor Corporation (NSC) were predicted accurately using the PWL waveform generated by the proposed test generation algorithm. To predict the remaining performance parameters, four PWL waveforms were designed using the designer's recommendations. The test stimuli were generated by the AWG of an Eagle mixed-signal tester, ETS-500D and applied to 587 ICs. The transient response of the circuits was sampled and analyzed for predicting the performance parameters. The key outcomes of this experiment were: 1) a ten times speed up in production testing; 2) simpler test configuration; 3) accurate prediction of the performance parameters; and 4) accurate classification of all the 587 ICs.

The remainder of this paper is organized as follows. In the next section, a multivariate parametric fault modeling methodology that forms the basis of the alternate testing technique, is discussed. In Section III a methodology for postprocessing the transient test measurements to extract the information about the performance parameters of the CUTs is discussed. In Section IV,

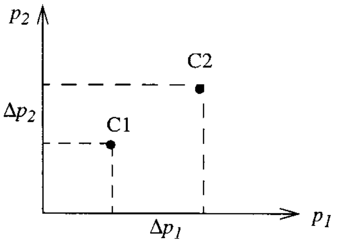


Fig. 1. A 2-D circuit parameter space.

the test generation method is discussed in detail. In Section V, the results from the experiments with the operational amplifier test vehicle are discussed. Finally, Section VI concludes the paper, with directions of future research.

II. Multivariate Parametric Fault Modeling

In this section, a multivariate parametric fault modeling methodology is described. The alternate testing technique presented in this paper is developed from the theoretical framework of this fault modeling methodology.

The performance of an analog circuit is determined by a set of associated *circuit parameters.* Gate oxide thickness of transistors, threshold voltages of NMOS and PMOS transistors, value of a lumped resistance, etc., are examples of circuit parameters. These circuit parameters can be denoted as in (1) where n_p is the total number of circuit parameters which affect the circuit performance. A circuit with n_p circuit parameters can be represented by a point in the n_p dimensional *circuit parameter space*

$$\boldsymbol{p} = [p_1, p_2, \dots, p_{n_p}]. \tag{1}$$

A two-dimensional (2-D) circuit parameter space with two circuits, C1 and C2, is shown in Fig. 1. In the figure, C1 is the *nominal fault free circuit* and C2 is a circuit with parametric deviations in the circuit parameters p_1 and p_2.

The quality of an analog circuit is evaluated by measuring various *performance parameters* which reflect its transient, frequency, and dc performance capabilities. Slew rate, gain bandwidth, etc. are examples of performance parameters of an opamp. These performance parameters are denoted as in (2) where n_s is the total number of performance parameters

$$\boldsymbol{s} = [s_1, s_2, \dots, s_{n_s}]. \tag{2}$$

A circuit in the circuit parameter space can be mapped to a point in the *performance parameter space* using a set of n_s mapping functions. These mapping functions are denoted as in (3) and are defined by the relation given in (4)

$$\boldsymbol{f}_{ps} = [f_{ps1}, f_{ps2}, \dots, f_{psn_s}] \tag{3}$$

$$f_{psi} : \boldsymbol{p} \rightarrow s_i, \boldsymbol{p} \in \Re^{n_p}, s_i \in \Re, i = 1 \dots n_s. \tag{4}$$

The performance parameters of a circuit have to satisfy certain *specifications* which are given by lower and/or upper bounds on the performance parameters. Specifications with a single bound are known as *single-ended specifications* and those with both upper and lower bounds are known as *double-ended specifications.* Note that a double-ended specification can be decomposed to two single-ended specifications. Let n_1 denote the number of performance parameters with single-ended specifications and n_2 denote the number of performance parameters with double-ended specifications, such that $n_1 + n_2 = n_s$. The total number of single-ended specifications after decomposing all the double-ended specifications is given by (5)

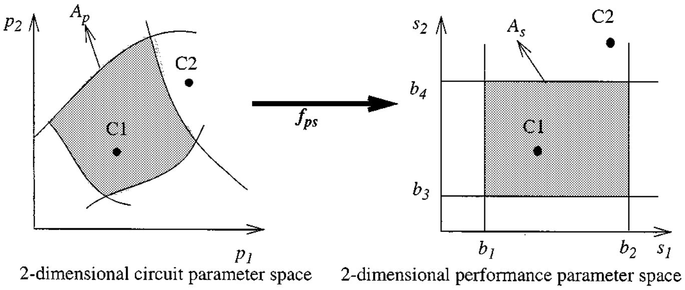

p_2
A_p
C2
C1
f_{ps}
p_1
2-dimensional circuit parameter space
s_2
A_s
C2
b_4
C1
b_3
b_1
b_2
s_1
2-dimensional performance parameter space

Fig. 2. Function f_{ps} maps circuit instances C1 and C2 from the 2-D circuit parameter space to the 2-D performance parameter space.

$$n_b = n_1 + 2n_2. \tag{5}$$

The lower or upper bound of the ith single-ended specification is denoted as b_i. Following the notation in [35], the ith single ended specification (on the performance parameter s_j) defines a region $A_{s,i}$ in the n_s–dimensional performance parameter space containing all the performance parameter values satisfying (6). In (6), different subscripts are used for performance parameter s and specification bound b, since a performance parameter can have more than one bound. Performance parameter sets satisfying all the n_b single-ended specifications form the *acceptance region in the performance parameter space* A_s defined by (7)

$$A_{s,i} = \left\{ s \in \Re^{n_s} \middle| \begin{matrix} s_j > b_i \text{ if lower bound} \\ \\ s_j < b_j \text{ if upper bound} \end{matrix} \right\} \tag{6}$$

$$A_s = \bigcap_{i=1,\dots,n_b} A_{s,i}. \tag{7}$$

Circuits with performance parameters lying in A_s are classified as *fault-free* or *good circuits.* Circuits with performance parameters lying outside A_s are *faulty* or *bad circuits.* Since the circuit and performance parameter spaces are related by the mappings given in (4), the acceptance region in the performance parameter space is implicitly related to the *acceptance region in the circuit parameter space* A_p as defined by (8) and (9)

$$A_{p,i} = \left\{ \boldsymbol{p} \in \Re^{n_p} \middle| \boldsymbol{f}_{ps}(\boldsymbol{p}) \in A_{s,i} \right\} \tag{8}$$

$$A_p = \bigcap_{i=1,\dots,n_b} A_{p,i}. \tag{9}$$

Fig. 2 shows 2-D circuit parameter and performance parameter spaces with a nominal fault-free circuit C1 and a circuit C2 with parametric fault. Functions $\boldsymbol{f_{ps}} = [f_{ps1}, f_{ps2}]$ map the two circuits from the circuit parameter space to the performance parameter space. Since these mappings are not generally available in closed form, they are evaluated using numerical circuit simulation. Performance parameters s1 and s2 have double-ended

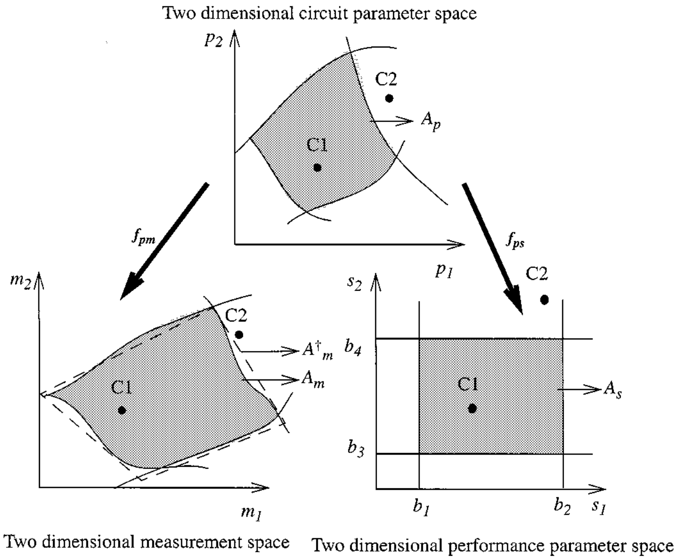

Two dimensional circuit parameter space
p_2
p_1
C2
C1
A_p
f_{pm}
f_{ps}
m_2
m_1
C2
C1
$A^\dagger_m$
A_m
s_2
s_1
b_4
b_3
b_1
b_2
C2
C1
A_s
Two dimensional measurement space
Two dimensional performance parameter space

Fig. 3. Two-dimensional circuit, performance and measurement parameter spaces.

specifications on them ($n_1 = 0, n_2 = 2$) and these specifications can be decomposed into four ($n_b = 4$) single-ended specifications ($s_1 > b_1$, $s_1 < b_2$, $s_2 > b_3$ and $s_2 < b_4$). Circuit C2 lies outside the acceptance region A_s in the performance parameter space and is a faulty circuit. Boundaries of the acceptance region in the performance parameter space are mapped back into the circuit parameter space to get the acceptance region in the circuit parameter space A_p. The two acceptance regions are shown as shaded areas in the figure.

Alternate test measurements are certain easy-to-perform (*vis. a. vis.* measuring the performance parameters) measurements by which the CUT can be classified as good or bad. Assuming there are n_m alternate test measurements, they can be represented as in (10)

$$\boldsymbol{m} = [m_1, m_2, \ldots, m_{n_m}]. \tag{10}$$

A circuit in the n_p–dimensional circuit parameter space can be mapped to a point in the n_m–dimensional *measurement space* using n_m mapping functions denoted as in (11) and defined by the relation in (12)

$$\boldsymbol{f_{pm}} = [f_{pm1}, f_{pm2}, \cdots, f_{pmn_m}] \tag{11}$$

$$f_{pmi} : \boldsymbol{p} \rightarrow m_i,\ \boldsymbol{p} \in \Re^{n_p}, m_i \in \Re, i = 1 \ldots n_m. \tag{12}$$

Every circuit within the acceptance region in the circuit parameter space can be mapped into the measurement space using the mappings given in 12 to give an *acceptance region in the measurement space* A_m defined by (13) and (14)

$$A_{m,i} = \left\{ \boldsymbol{m} = \boldsymbol{f_{pm}}(\boldsymbol{p}) \in \Re^{n_m} \middle|_{\boldsymbol{p} \in A_{p,i}} \right\} \tag{13}$$

$$A_m = \bigcap_{i=1,\ldots,n_b} A_{m,i}. \tag{14}$$

Fig. 3 shows a 2-D measurement space along with the circuit parameter space and the performance parameter space. The acceptance region in all three spaces is shown as shaded areas. Mapping from the circuit parameter space to the measurement space is usually done through numerical circuit simulation.

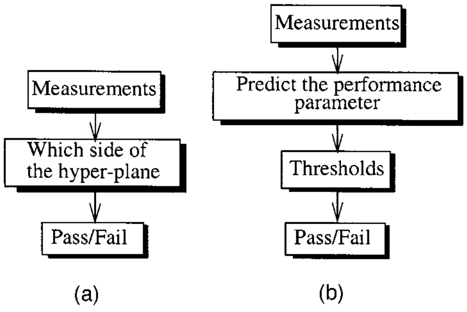

Measurements
Which side of the hyper-plane
Pass/Fail
(a)
Measurements
Predict the performance parameter
Thresholds
Pass/Fail
(b)

Fig. 4. (a) Previous testing approach. (b) Proposed testing approach.

Test criteria are certain conditions on the alternate test measurements based on which a circuit is classified as faulty or fault-free during alternate testing. Assuming there are n_t test criteria on the alternate test measurements $\boldsymbol{m}$, they can be represented as in (15), where t_i are certain specified bounds

$$f_i(\boldsymbol{m}) > t_i \text{ or } f_i(\boldsymbol{m}) < t_i, i = 1 \ldots n_t. \tag{15}$$

Each of the test criterion defines a region in the measurement space defined by (16).The region in the measurement space that satisfies all the test criteria is defined by (17)

$$A^\dagger_{m,i} = \left\{ \boldsymbol{m} \in \Re^{n_m} \middle| \begin{matrix} f_i(\boldsymbol{m}) > t_i \text{ if lower bound} \\ \\ f_i(\boldsymbol{m}) < t_i \text{ if upper bound} \end{matrix} \right\} \tag{16}$$

$$A^\dagger_m = \bigcap_{i=1,\ldots,n_b} A^\dagger_{m,i}. \tag{17}$$

In [1] and [31]–[34], a hyperplane was derived for each of the single-ended specifications ($n_t = n_b$) as the test criterion. Four such hyperplanes are shown in Fig. 3. It can be seen that the acceptance region in the measurements space A_m cannot be accurately bounded with hyperplanes. This leads to a misclassification of circuits during alternate testing using the techniques described in [1] and [31]–[34]

III. Predicting Performance Parameters From Transient Response

Given a set of test measurements, determining the test criteria involves finding the boundaries of the acceptance region in the measurement space. Previous methods [1], [31]–[34] approximated the boundaries of the acceptance region in the measurement space using hyperplanes. Their methodology is depicted in Fig. 4(a). For general nonlinear analog circuits the boundaries of the acceptance region in the measurement space could be very complex and simple approximations using hyperplanes could lead to large misclassifications. Our methodology for determining test criteria is shown in Fig. 4(b). We propose to calculate the performance parameters of the circuit from the transient test measurements. Circuits are then classified by comparing the calculated performance parameters with the specification bounds.

To predict n_s performance parameters from the transient test measurement, it is necessary to determine n_s mapping functions relating the measurements to the n_s performance parameters.These mapping functions are denoted as in (18)

$$\boldsymbol{f_{ms}} = [f_{ms1}, f_{ms2}, \ldots, f_{msn_s}]. \tag{18}$$

These functions are to be derived such that

$$\boldsymbol{f_{ps}}(\boldsymbol{p}) = \boldsymbol{f_{ms}}(\boldsymbol{f_{pm}}(\boldsymbol{p})) \text{ for } (\forall \boldsymbol{p} \in \Re^{n_p}). \tag{19}$$

If the postprocessing function $\boldsymbol{f_{ms}}$ satisfies the relation given in (19), then $\boldsymbol{f_{ms}}(\boldsymbol{m})$ becomes equal to the performance parameters s. As a consequence, the specification bound b_i itself becomes the ith test criteria. Using the notation presented in Section II, ith test criteria can be defined as in (20). Note that, in (20), different subscripts are used for the calculated performance parameters and the corresponding bounds as there can be performance parameters with double-ended specifications. Thus there are n_b such test criteria on n_s calculated performance parameters and if all of them are satisfied, the CUT is declared as fault-free

$$\text{CUT is fault-free if} \begin{cases} f_{msj}(\boldsymbol{m}) > b_i & \text{if upper bound} \\ f_{msj}(\boldsymbol{m}) < b_i & \text{if lower bound.} \end{cases} \tag{20}$$

Thus we have converted a problem of accurate boundary determination to one of function approximation. The key advantages of this technique are enumerated below.

1) Since $\boldsymbol{f_{ms}}(\boldsymbol{m})$ tracks the performance parameter values, it is possible to determine directly from $\boldsymbol{f_{ms}}(\boldsymbol{m})$ whether the CUT meets its specifications. Postprocessed measurements $\boldsymbol{f_{ms}}(\boldsymbol{m})$ contain much more information about the performance parameters than the measurement themselves.
2) There are robust nonlinear function approximation techniques which can be used to derive mapping functions $\boldsymbol{f_{ms}}(\boldsymbol{m})$. These functions can capture highly nonlinear relations between measurements and performance parameters. Below we discuss a multivariate regression technique to derive the mapping functions.

A. Deriving the Functions $\boldsymbol{f_{ms}}$

The functions $\boldsymbol{f_{ps}}$ and $\boldsymbol{f_{pm}}$ are very complex and are not available in closed form. So it is not possible to directly derive the functions $\boldsymbol{f_{ms}}$ using the condition in (19). Hence, we resort to function approximation using regression which has a robust framework in the realm of statistics. The technique used for approximating the functional mapping between measurements and performance parameters needs to have the following desirable properties [36], [37].

1) It should be able to approximate highly nonlinear functions accurately.
2) It should be able to handle large dimensionality of dependent variables.
3) It should be immune to the problem of overfitting.

Multivariate adaptive regression splines (MARS) [38] is a tool which has the above-mentioned desirable properties. Hence we use MARS to derive the postprocessing functions $\boldsymbol{f_{ms}}$. One can also use neural networks or any other regression strategy to approximate the function $\boldsymbol{f_{ms}}$. For a detailed description of MARS algorithm refer to [38]. Below we summarize the steps involved in deriving the mapping $\boldsymbol{f_{ms}}$.

1) Perform both the conventional specification-based tests and the transient tests on an initial set of ICs coming out of the production line. These ICs constitute the *training set* for building the MARS models relating the transient test measurements and performance parameters.
2) Use the transient test measurement data (as the independent variables) and the performance parameters (as the dependent variables) of the circuits in the training set to find the mapping $\boldsymbol{f_{ms}}$ using the MARS procedure.
3) Store the regression splines in the tester computer for future use.

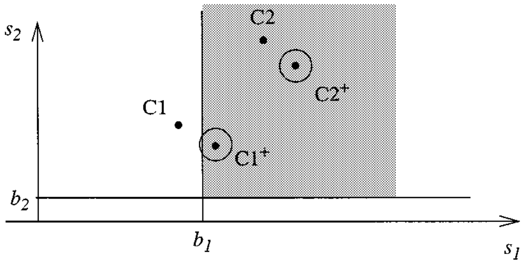


Fig. 5. Circuits at the boundary are more prone to misclassification.

The function $\boldsymbol{f_{ms}}$ derived using MARS will not be able to predict the performance parameters of the CUT accurately. The inaccurate prediction of the performance parameters will lead to misclassification of circuits. Two chief reasons for misclassification are: 1) transient test measurement errors and 2) errors in the approximation for $\boldsymbol{f_{ms}}$. The chances of misclassification can be minimized by deriving an appropriate transient test stimulus to excite the circuit. In the following section a methodology for deriving PWL transient test stimulus is discussed.

IV. PWL Test Stimulus Generation for Minimizing the Chances of Misclassification

Generating the best PWL test stimulus is an optimization problem. Two components of this optimization problem are: 1) a fitness function for guiding the optimization and 2) an optimization procedure to find the PWL waveform with maximum fitness. In the following section, the fitness function for guiding the optimization is derived. Maximizing this fitness function guarantees that a minimum number of circuits gets misclassified during alternate testing. In Section IV-B, a genetic algorithm-based search procedure is described.

A. Deriving a Fitness Function

Circuits with performance parameters close to the specification boundary are affected severely by the nonidealities in the test procedure and are more prone to the problem of misclassification. This can be explained with the help of Fig. 5. Fig. 5 shows a 2-D performance parameter space and two specification bounds b_1 and b_2. Circuit C1 is a faulty circuit close to the specification boundary and C2 is well within the acceptance region. $C1^+$ and $C2^+$ represent the performance parameters predicted using $\boldsymbol{f_{ms}}$. These do not coincide with the actual performance

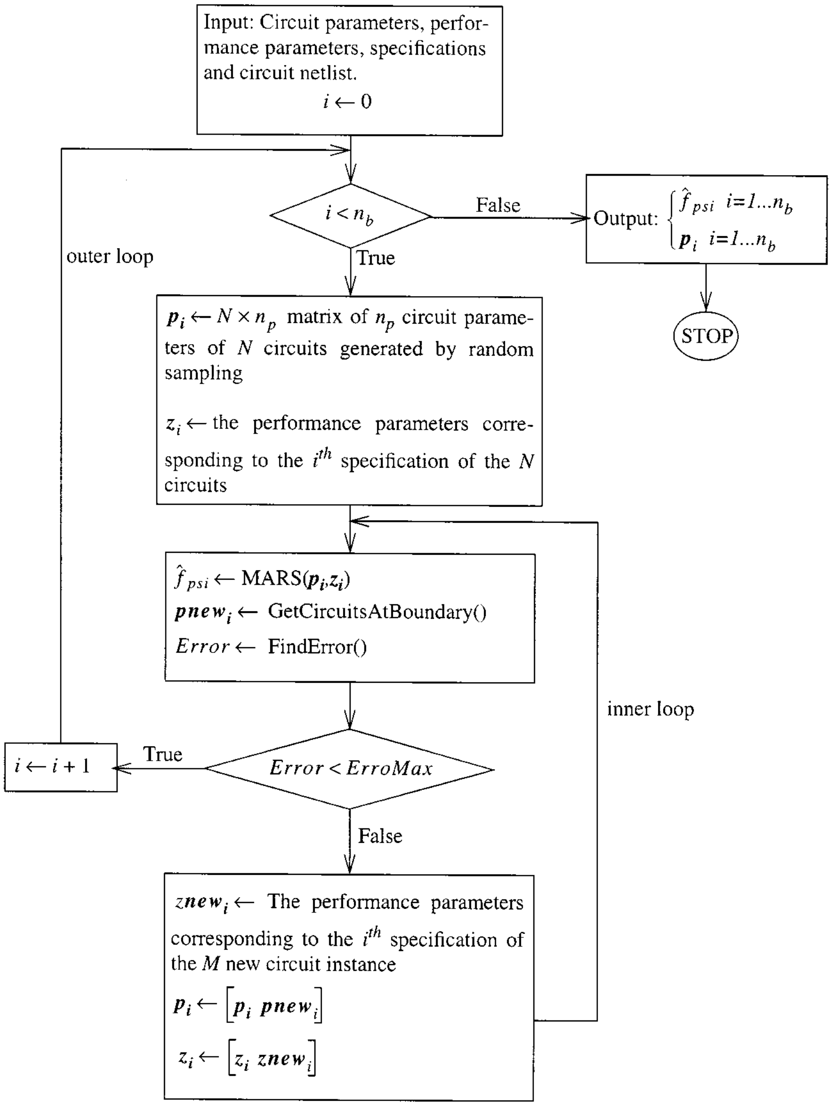

Input: Circuit parameters, performance parameters, specifications and circuit netlist.
$i \leftarrow 0$
$i < n_b$
False
Output: $\hat{f}_{psi}$ $i=1...n_b$, p_i $i=1...n_b$
STOP
outer loop
True
$p_i \leftarrow N \times n_p$ matrix of n_p circuit parameters of N circuits generated by random sampling
$z_i \leftarrow$ the performance parameters corresponding to the i^{th} specification of the N circuits
$\hat{f}_{psi} \leftarrow$ MARS(p_i,z_i)
$pnew_i \leftarrow$ GetCircuitsAtBoundary()
$Error \leftarrow$ FindError()
inner loop
$i \leftarrow i+1$
True
$Error < ErroMax$
False
$znew_i \leftarrow$ The performance parameters corresponding to the i^{th} specification of the M new circuit instance
$p_i \leftarrow [p_i \; pnew_i]$
$z_i \leftarrow [z_i \; znew_i]$

Fig. 6. Flow diagram for GenerateCriticalCircuits().

parameters represented by C1 and C2 due to an error in function approximation f_{ms}. Also, the circles around $C1^+$ and $C2^+$ show the inaccuracy in the prediction of the performance parameters due to measurement errors. It can be seen that the circuit C1 has a high probability of getting misclassified while C2 is classified correctly even with an inaccurate f_{ms} and measurement errors. The circuits close to the boundary are henceforth referred to as *critical circuits.* The fitness function should focus on the critical circuits.

Let $C(i)_{\text{fault-free}}$ be a circuit within the acceptance region close to a circuit at the boundary $C(i)$ and let $C(i)_{\text{faulty}}$ be a circuit outside the acceptance region close to $C(i)$. Let n_m measurements be made on the transient response of both $C(i)_{\text{fault-free}}$ and C$(i)_{\text{fautly}}$. Also let $m(i,j)_{\text{fault-free}}$ be the jth measurement made on $C(i)_{\text{fault-free}}$ and $m(i,j)_{\text{faulty}}$ be the jth measurement made on $C(i)_{\text{faulty}}$. If n_c critical circuits are used, the fitness function driving the search for optimum PWL transient stimulus is given in (21). Maximizing this fitness function guarantees that the change in the transient response of the circuit due to process variation is maximized for all the circuits close to acceptance boundary

$$Fitness = \sum_{j=1}^{n_m} \sum_{i=1}^{n_c} \left| m(i,j)_{\text{faulty}} - m(i,j)_{\text{faulty-free}} \right|. \quad (21)$$

To calculate the fitness function for a given PWL test stimulus, it is necessary to have the circuit parameters of a set of critical circuits. The problem of determining the circuit parameters of a set of critical circuits can be stated as follows: *Given* the circuit parameters, performance parameters, specifications, and the circuit netlist, *find* a set of circuits (critical-circuits) in the circuit parameter space whose performance parameters are close to the specification bounds. This problem is complicated because the acceptance boundaries are specified in the performance parameter space and the relations between circuit parameters and performance parameters are very complex even for simple analog circuits.

Critical circuits can be generated by randomly selecting circuits in the circuit parameter space and simulating these circuits to see if the performance parameters of these circuits are close to the specification bounds. This method is based on a random search and a set of critical circuits is obtained only after performing a large number of circuit simulations. We employ two techniques to cut down the simulation time for deriving the critical circuits: 1) An iterative directed search is used instead of random search and 2) The relations between the circuit parameters and performance parameters are modeled using MARS [38] to cut down the number of circuit simulations [36], [37].

The flow diagram for generating critical circuits using the iterative procedure is shown in Fig. 6. A set of N circuits is generated by uniformly sampling the circuit parameter space and the n_p circuit parameters of these circuits are stored in an $N \times n_p$ matrix $\boldsymbol{p_i}$. These N circuits are then simulated and the performance parameters corresponding to the ith single-ended specification are evaluated and stored in an $N \times 1$ matrix $\boldsymbol{z_i}$. A MARS model $\hat{f}_{psi}$ is built with the circuit parameters $\boldsymbol{p_i}$ as the independent variable and the performance parameters $\boldsymbol{z_i}$ as the dependent variable. Using this MARS model, a set of M critical circuits corresponding to the ith single-ended specification is generated by the procedure GetCircuitsAtBoundary() and these circuits are stored in an $M \times n_p$ matrix $\boldsymbol{pnewi}$. In the procedure GetCircuitsAtBoundary(), critical circuits are generated by binary search. Two circuit instances at either side of the specification bound bi are used as the initial points for the binary search. Note that during this binary search, performance parameters are evaluated using the MARS model $\hat{f}_{psi}$ to avoid costly circuit simulation. The M newly generated circuits are used by procedure FindError() to calculate the average squared error between the performance parameters predicted by $\hat{f}_{psi}$ and those obtained through actual simulation. If this error is less than a threshold, $\hat{f}_{psi}$ is declared accurate near the specification bound and a desired number of critical circuits are generated using. If the error is more than the threshold, the set of M circuits are added to the training set for the MARS routine and $\hat{f}_{psi}$ is rebuilt. Thus, the accuracy of the MARS model is improved by iterating through the inner loop of the procedure. MARS models corresponding to all the single-ended specifications are generated by iterating through the outer loop. The outputs of this procedure are the MARS models relating the circuit performance parameters to the process parameters and a set of critical circuits.

The quality of the fitness function defined in (21) depends on the number of critical circuits n_c in the equation. It is better to use a maximum possible number of critical circuits while evaluating the fitness of a transient stimulus. However, having a greater number of critical circuits will increase the simulation time and the overall test generation time.

B. Genetic Algorithm for Test Generation

Since a large number of voltage levels can be generated by the AWG for each corner point of the PWL test stimulus, the search space for finding the optimum PWL test stimulus is very large. For such problems genetic algorithms have shown the ability to move toward better solutions by selecting possible solutions from a large search space [39], [40]. Hence, genetic algorithms are used to search for the optimum PWL transient stimulus. An overview of the test generation procedure based on genetic algorithms is shown in Fig. 7.

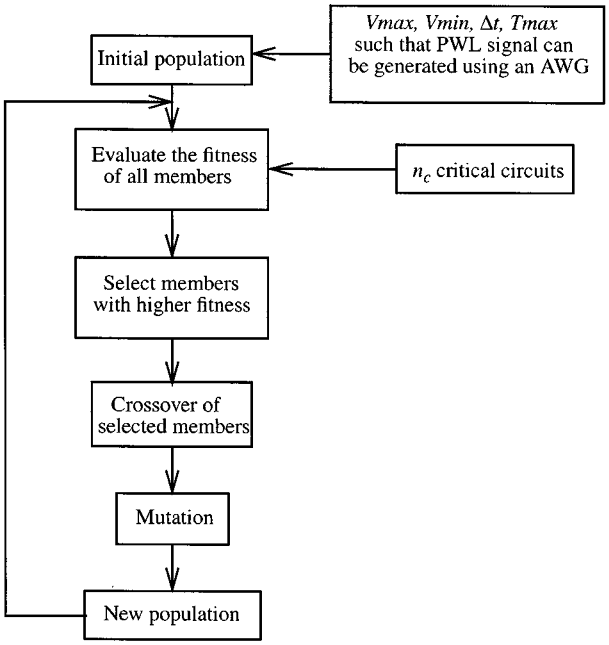

Initial population
Vmax, Vmin, Δt, Tmax such that PWL signal can be generated using an AWG
Evaluate the fitness of all members
n_c critical circuits
Select members with higher fitness
Crossover of selected members
Mutation
New population

Fig. 7. Overview of the genetic search.

Unlike other search procedures where the search is done on the parameters themselves, the genetic algorithm works by coding the parameter set into genetic strings. The parameters of the desired PWL test stimulus, namely maximum voltage swing $V\max$, minimum voltage swing $V\min$, the time between the corner points of the PWL test signal Δt, and the maximum testing time $T\max$, are taken as inputs to the test generation program. These parameters can be controlled appropriately to make sure that the PWL test signal can be accurately generated using the AWG. Each member (chromosome) of the population has $\lfloor Tmax/\Delta t \rfloor$ genes in them. The ith gene of the genetic chromosome is an integer representing the voltage at time point t_i given by (22)

$$t_i = i \cdot \Delta t. \tag{22}$$

Let the maximum allowed integer value of a gene be $k\max$ and let the ith gene have a value k, then the PWL transient waveform that has a voltage v_i at the corner point t_i is given by

$$v_i = V\min + \frac{(V\max - V\min)}{k\max} \cdot k. \tag{23}$$

An example of the encoding is shown in Fig. 8 with $V\max = 5V$, $V\min = -5V$, $\Delta t = 1$ ms, $T\max = 10$ ms. There are ten corner points for this PWL waveform and hence there are ten genes in the genetic chromosomes. The values of the genes (k) and the shape of the PWL waveform are shown in the figure.

Initially, a population of PWL test stimuli are generated at random and the fitness function of all the members of the population are evaluated through simulation. A set of n_c critical circuits generated by the procedure GenerateCriticalCircuits(), (Fig. 6) is taken as input to the procedure for evaluating the fitness. From these circuits, a set of $2n_c$ circuits at either side of

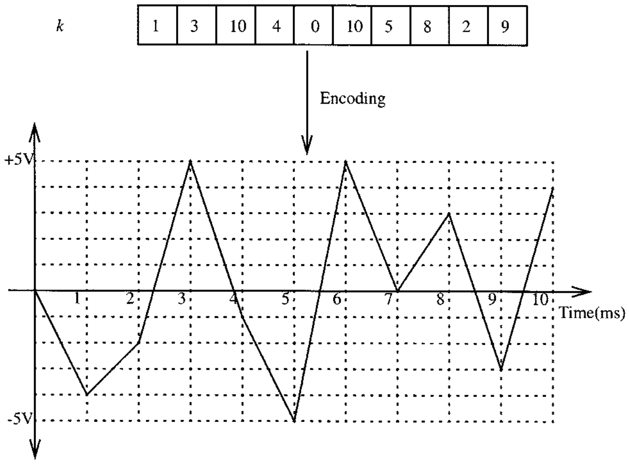


Fig. 8. Encoding a PWL waveform as a genetic string.

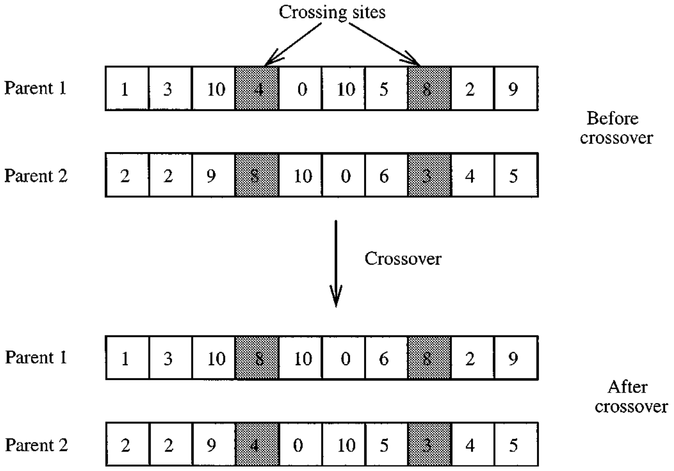

Crossing sites
Parent 1 1 3 10 4 0 10 5 8 2 9
Parent 2 2 2 9 8 10 0 6 3 4 5
Before crossover
Crossover
Parent 1 1 3 10 8 10 0 6 8 2 9
Parent 2 2 2 9 4 0 10 5 3 4 5
After crossover

Fig. 9. Uniform crossover.

the acceptance boundary is generated. These circuits are simulated with the PWL waveforms of the population and the fitness of each of the members is evaluated using (21).

A set of selected members of the population with high fitness values is taken and subjected to crossover. The selection of strings for crossover is biased toward strings having the highest fitness value so that the average fitness of successive populations tends to increase. We used *tournament selection* for selecting the parents for reproduction. *Tournament selection* involves picking two strings from the population and selecting the better for reproduction.

The crossover operator takes genes from each of the parent strings and combines them to create child strings. We used the *uniform crossover* scheme for creating child strings. Fig. 9 shows how *uniform crossover* is performed to produce the child strings. Each gene of the parent strings is chosen with certain probability and is swapped to yield the two child strings.

After the child strings are created, the genes of the child strings can undergo mutation. For mutation, a gene is selected with a certain probability (*mutation probability*) and is replaced with a random number within the allowed range. Thus, after selection, crossover, and mutation, the resulting new population undergoes the entire cycle of genetic evolution. With each of the iteration, the average fitness of the members increases and after a large number of iterations the fitness does not improve much. At that point the PWL waveform with highest fitness is selected as the optimum PWL test waveform.

TABLE I
PERFORMANCE PARAMETERS AND SPECIFICATIONS

Number	Performance parameter	Specification
1	Supply current (Isup)	< 1.5mA
2	Short circuit current (Sourcing) (Isc_src)	> 30mA
3	Short circuit current (sinking) (Isc_snk)	> 30mA
4	Slew rate (SR)	> 0.5V/µs
5	Offset voltage (Vos)	< 3mV

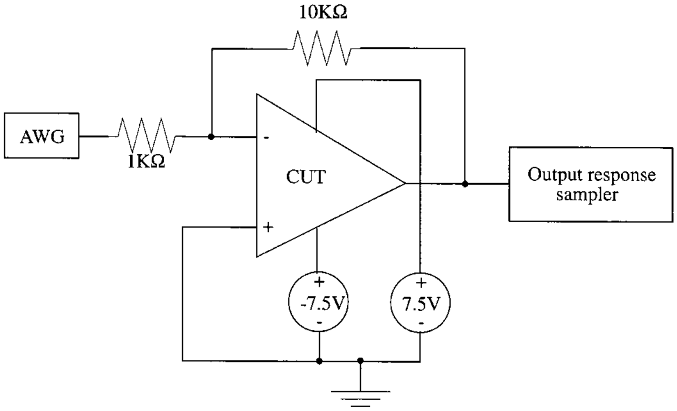


Fig. 10. Test configuration for the low-power opamp.

V. EXPERIMENTS WITH A LOW-POWER OPAMP

The testing methodology presented in this paper was experimentally validated in a production environment using a low-power operational amplifier from National Semiconductor Corporation as the test vehicle. To validate the testing methodology, 587 packaged ICs were taken out from the production line. The list of targeted performance parameters and the specifications are listed in Table I. All the performance parameters were measured at 15 V power supply conditions. For sourcing tests (Isc_src) the CUT output was forced to source load current and for the sinking tests (Isc_snk), the CUT output was forced to sink the load current. The derived transient test was applied to all the 587 ICs using an Eagle mixed-signal tester, ETS-500D. The existing test program was using a Teradyne tester and hence conventional specification tests were performed using this Teradyne tester. Results of these experiments are summarized in this section.

A. PWL Test Stimulus Generation

With the help of circuit designers and test engineers, a PWL test stimulus was generated for the test vehicle in a computer-aided environment. The following were the steps involved in the transient test generation.

1) The level-2 SPICE netlist of the circuit was obtained from the circuit designer.
2) A set of performance parameters, that are to be replaced with the alternate tests, were selected from the data sheet

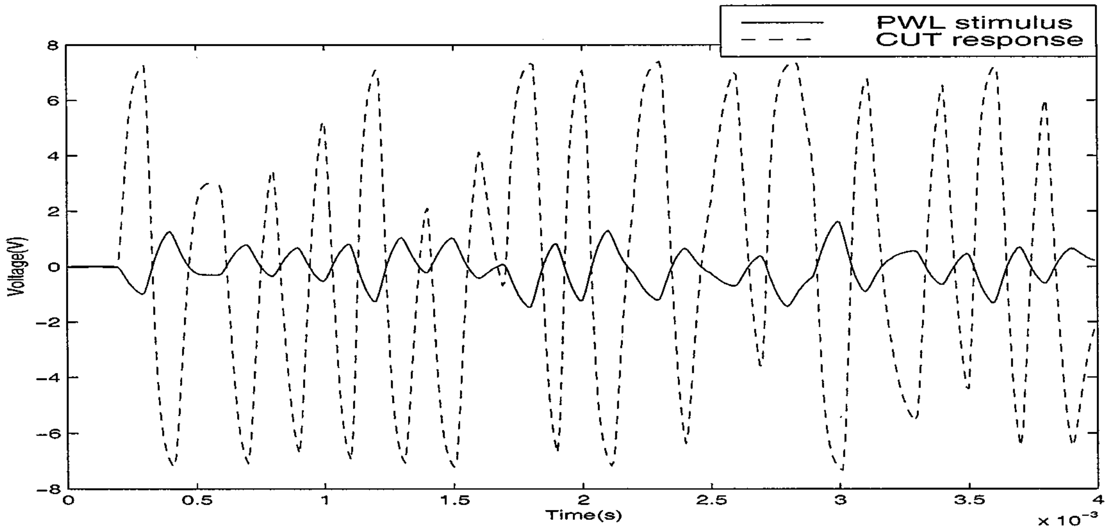


Fig. 11. PWL test stimulus for the LMC7101 and CUT response.

TABLE II
STANDARD DEVIATION OF THE RESIDUALS

	Standard deviation of the residuals					
Size of the training set	Supply current (mA)	Short circuit current (sourcing) (mA)	Short circuit current (sinking)(mA)	Slew rate (v/μs)	Offset voltage (mV)	CPU time (s)
100	0.0187	1.1353	1.041	0.0132	0.3582	161.12
200	0.0164	0.9199	0.9582	0.0099	0.3622	364.95
300	0.0223	0.6924	0.8147	0.0099	0.3351	611.21
400	0.0136	0.6920	0.8748	0.0089	0.3169	826.27
500	0.0119	0.7640	0.7689	0.0077	0.3168	960.73

of the device. Table I shows the five performance parameters of the opamp targeted during the alternate testing process.

3) With the help of designers, a set of important circuit parameters for the test vehicle was identified from the circuit's level-2 SPICE netlist. A circuit parameter is said to be important if the selected performance parameters are highly sensitive to the circuit parameters of the CUT. There were 29 important circuit parameters for the test vehicle.
4) An appropriate test configuration for the CUT was decided with the help of circuit designers. Since open-loop configuration of the opamp is highly sensitive to the nonidealities of the test environment, it was decided that the opamp should be connected in a negative feedback configuration. When a unity feedback configuration was used, the circuit response was insensitive to the parametric deviations in the process parameters. Hence, it was decided that the CUT must be connected in a negative feedback configuration with a gain of 10 during transient testing. Fig. 10 shows the final test configuration of the opamp.
5) Since the maximum power supply swing for the test vehicle was 15 V and the feedback gain was 10, it was decided that $V\max = +2\ V$ and $V\min = -2\ V$ for the PWL test stimulus.
6) Using the data sheet of the Eagle tester, ETS–500D, it was decided that the time duration between the corner points of the PWL test stimulus (Δt) must be at least 0.1 ms. When a shorter time duration was used, the tester was not able to reproduce the PWL test waveform accurately. Also, a maximum testing time of 4 ms was selected.
7) Circuit netlist, performance parameters, specifications, a list of important circuit parameters, and the parameters of the PWL test waveform were used to generate the optimum PWL test stimulus. The arbitrary waveform generator (AWG) of the Eagle tester ETS-500D was programmed to generated the PWL test stimulus. The PWL test stimulus generated by the tester and the CUT response are shown in Fig. 11. The transient response of the CUT was sampled 400 times and these 400 transient test measurements were used for fault detection.

B. Deriving f_{ms}

Before using the transient test in production, it is necessary to derive the functions f_{ms} relating the transient test measurements to the five performance parameters. The accuracy of these functions depends on the size of the training set used for deriving them. Residuals of the prediction, r, are defined in (24).

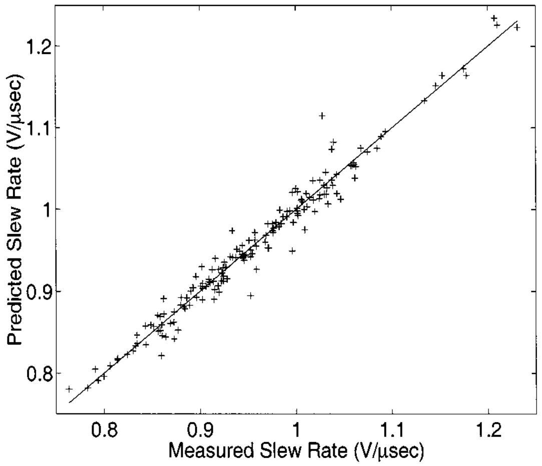


Fig. 12. Comparison of predicted slew rate with slew rate measured using a Teradyne tester.

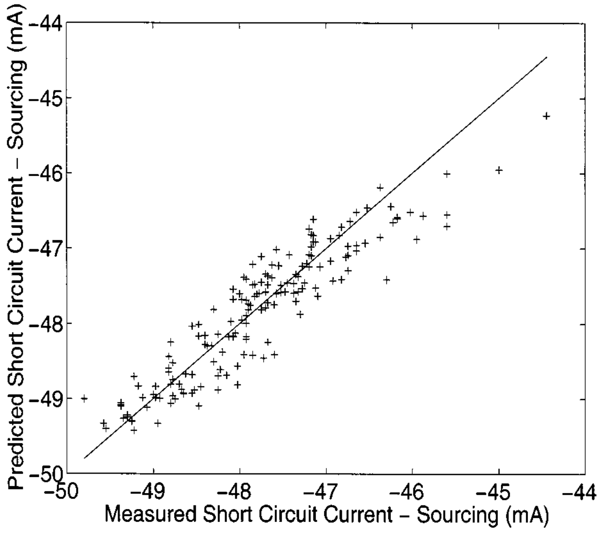


Fig. 13. Comparison of predicted short circuit current sourcing with short circuit current sourcing measured using a Teradyne tester.

Assuming n circuits are used for evaluating the accuracy of prediction, $\boldsymbol{r}$ is an $n \times n_s$ matrix

$$\boldsymbol{r} = \boldsymbol{f_{ms}}(\boldsymbol{m}) - \boldsymbol{s}.$$

The effect of the size of the training set on the accuracy of performance parameter prediction and the CPU time for obtaining the function $\boldsymbol{f_{ms}}$ are summarized in Table II. Training sets of different sizes were used to derive $\boldsymbol{f_{ms}}$ and the residuals of the prediction was calculated for 80 ICs. These 80 ICs selected for validation were different from those in the training set. Table II gives the standard deviation of the residuals for all five performance parameters. It can be seen that the accuracy of prediction improves with the size of the training set. Also, the CPU time for building $\boldsymbol{f_{ms}}$ increases approximately linearly with the size of the training set. However, this is a one-time cost and does not affect the production test time. The best strategy here is to increase the size of the training set until the desired accuracy of prediction is achieved.

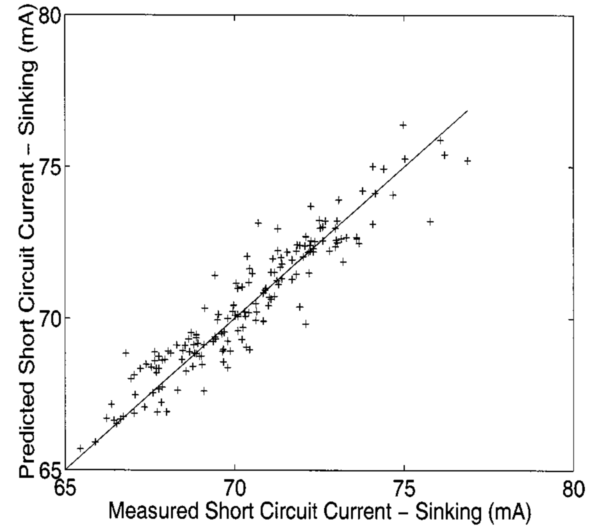


Fig. 14. Comparison of predicted short circuit current sinking measured using a Teradyne tester.

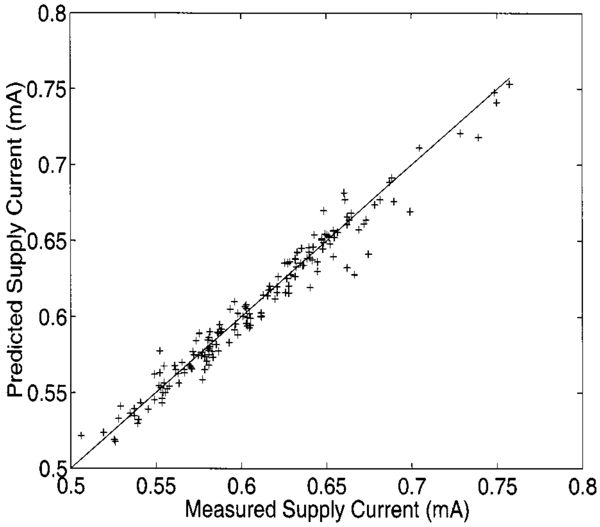


Fig. 15. Comparison of predicted supply current with supply current measured using a Teradyne tester.

C. Predicting the Performance Parameters

For all the ICs coming out of production line after deriving $\boldsymbol{f_{ms}}$, conventional specification tests are not performed, only the transient tests are performed. In the experiments with the low-power opamp, a training set of 300 ICs was used to derive $\boldsymbol{f_{ms}}$. For the remaining 287 ICs the five performance parameters were predicted from the transient test measurements using $\boldsymbol{f_{ms}}$. Fig. 12 shows the scatter plot of the slew rate with the measured slew rate on the X-axis and predicted slew rate on the Y-axis. A similar analysis for the other four performance parameters is depicted in Figs. 12–16. From this analysis it can be seen that the performance parameters can be predicted very accurately from the transient test measurements. Thus conventional specification-based tests can be completely eliminated.

D. Predicting the Remaining Performance Parameters

When trying to predict the remaining [(6)–(8)] performance parameters given in Table III, it was found that all four nodes of

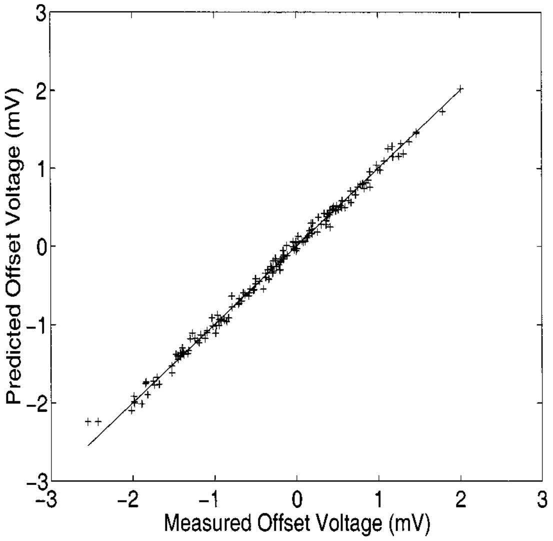


Fig. 16. Comparison of predicted offset voltage with offset voltage measured using a Teradyne tester.

TABLE III
REMAINING PERFORMANCE PARAMETERS

Number	Performance parameter	Specification
6	Common mode rejection ratio (CMRR)	> 65dB
7	Positive power supply rejection ratio (PPSRR)	> 65dB
8	Negative power supply rejection ratio (NPSRR)	> 65dB

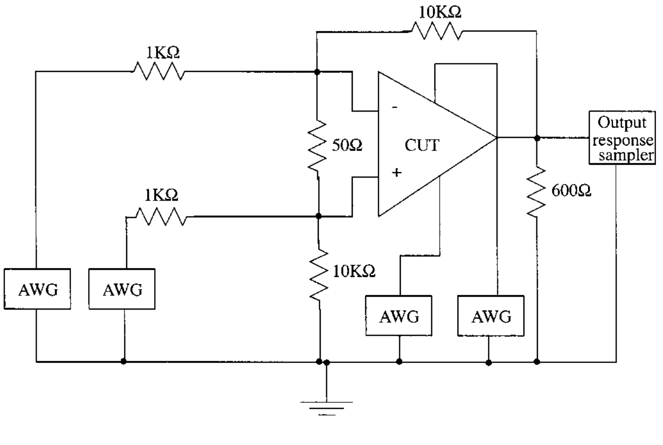


Fig. 17. Test configuration for the transient test for predicting performance parameters 6–10.

the opamp should be excited using PWL waveforms to get accurate prediction. Generating the 4-PWL linear waveforms will be computationally very expensive. Hence, the input waveforms were selected based on designer's recommendations. The test configuration for this transient test is shown in Fig. 17. The input transient test waveforms are shown in Fig. 18. An AWG is connected to all four inputs of the opamp and PWL test stimuli are applied to all these inputs. The predicted performance parameters are compared with the measured performance parameters in Figs. 19–21. It can be seen that the performance parameters can be accurately predicted from the transient test measurements.

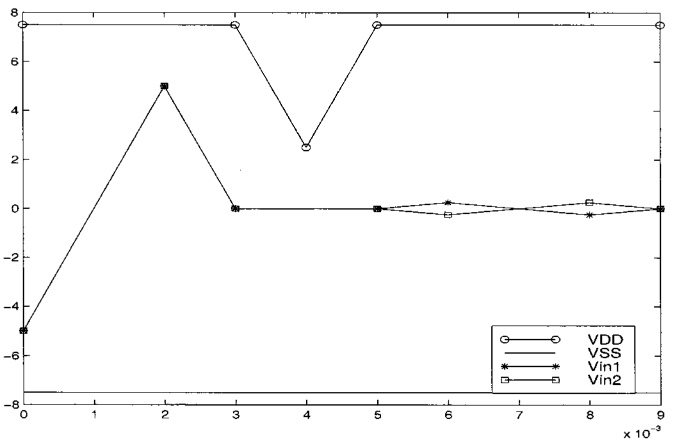


Fig. 18. PWL test stimuli for tracking the remaining specification.

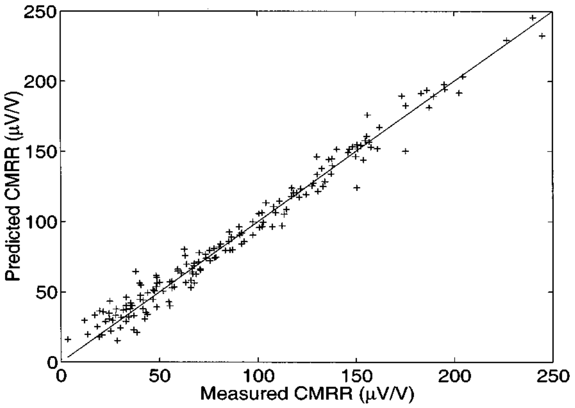


Fig. 19. Comparison of predicted CMRR with CMRR measured using a Teradyne tester.

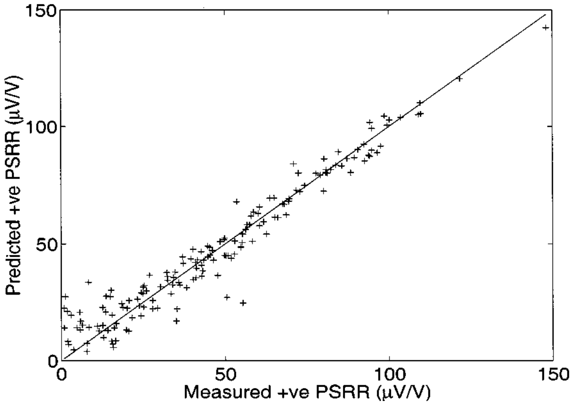


Fig. 20. Comparison of predicted positive PSRR with positive PSRR measured using aTeradyne tester.

E. Test Time Considerations

The total test stimulus application time for the transient test is 13 ms. The average CPU time for predicting all eight performance parameters from the transient test measurement data is 0.4 ms. Thus the total test time without taking the instrument and DUT setup time into account is 13.4 ms. The conventional specification test time for these eight specifications is well above 100 ms. Thus approximately an order of magnitude

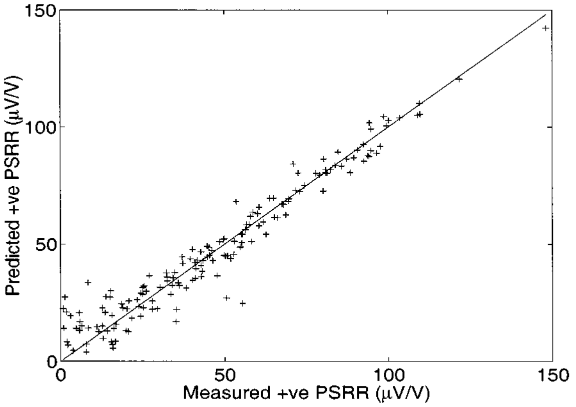


Fig. 21. Comparison of predicted negative PSRR with negative PSRR measured using a Teradyne tester.

speedup in production testing can be achieved by the proposed testing method.

VI. Conclusion

In digital circuits, customers are satisfied with a 97% fault coverage number. Customers of complex analog device ask for the specifications of the device and it is hard to satisfy them with a fault coverage number. With the increasing device complexity, testing for all these specifications is becoming a major bottleneck in reducing the production cost. This paper proposed fast transient testing as a viable alternative to the costly specification-based testing to bring down the production test cost.

The test methodology proposed in this paper was experimentally validated in a production environment. A low-power operational amplifier from National Semiconductor Corporation was used as the test vehicle. The postprocessed measurement obtained from the transient tests was able to track the performance parameters accurately. The total test time for the transient testing was 13.5 ms and all the ICs were classified correctly with the proposed test methodology. Using the conventional specification-based testing, the overall test time for this device is more than 100 ms. Thus the proposed method provides approximately an order of magnitude speed up in production testing over the existing method.

Acknowledgment

A patent application was filed with the U.S. Patent and Trademark Office on May 19, 1999 for this work.

References

[1] W. M. Lindermeir, "Design of robust test criteria in analog testing," in *Int. Conf. Computer Aided Design*, 1995, pp. 604–611.

[2] A. Walker, W. Alexander, and P. K. Lala, "Fault diagnosis in analog circuits using element modulation," *IEEE Design Test Computers*, pp. 19–29, Mar. 1992.

[3] "Semiconductor Industry Technology Workshop Conclusions," Semiconductor Industry Association, 1993.

[4] *European Design and Test Conf. Conclusions*, Mar. 1996.

[5] J. B. Brockman and S. W. Director, "Predictive subset testing: Optimizing IC parametric performance testing for quality, cost and yield," *IEEE Trans. Semiconduct. Manuf.*, vol. 2, pp. 104–113, 1989.

[6] W. Maly and Z. Pizlo, "Tolerance assignment for IC selection tests," *IEEE Trans. Computer-Aided Design*, pp. 156–162, Apr. 1985.

[7] T. M. Souders and G. N. Stenbakken, "Cutting the high cost of testing," *IEEE Spectrum*, pp. 48–51, Mar. 1991.

[8] G. N. Stenbakken and T. M. Souders, "Test-point selection and testability measures via QR factorization of linear models," *IEEE Trans. Instrum. Measurement*, pp. 406–410, June 1987.

[9] S. D. Huss and R. S. Gyurcsik, "Optimal ordering of analog IC tests to minimize test costs," in *Proceedings, Design Automation Conf.*, 1991, pp. 494–499.

[10] L. Milor and A. L. Sangiovanni-vincentelli, "Minimizing production test time to detect faults in analog circuits," *IEEE Trans. Computer Aided Design*, vol. 13, pp. 796–813, 1994.

[11] H. Walker and S. W. Director, "VLASIC: A catastrophic fault yield simulator for integrated circuits," *IEEE Trans. Computer Aided Design*, pp. 541–556, Oct. 1986.

[12] L. Milor and V. Viswanathan, "Detection of catastrophic faults in analog integrated circuits," *IEEE Trans. Computer Aided Design*, vol. 8, pp. 114–130, 1989.

[13] M. J. Marlett and J. A. Abraham, "DC-IATP: An iterative analog circuit test generation program for generating DC single pattern tests," in *Proc. IEEE Int. Test Conf.*, 1988, pp. 839–845.

[14] G. Devarayanadurg and M. Soma, "Analytical fault modeling and static test generation for analog IC's," in *Int. Conf. Computer Aided Design*, 1994, pp. 44–47.

[15] P. N. Variyam, J. Hou, and A. Chatterjee, "Test generation for analog circuits using partial numerical simulation," in *Int. Conf. VLSI Design*, 1999, pp. 597–602.

[16] N. B. Hamida and B. Kaminska, "Analog circuit testing based on sensitivity computation and new circuit modeling," in *Int. Test Conf.*, 1993, pp. 331–343.

[17] ——, "Multiple fault analog circuit testing by sensitivity analysis," *J. Electronic Testing*, vol. 4, pp. 331–343, 1993.

[18] N. Nagi, A. Chatterjee, A. Balivada, and J. A. Abraham, "Fault-based automatic test generator for linear analog devices," in *Proc. Int. Conf. Computer Aided Design*, 1993, pp. 88–91.

[19] M. Slamani, B. Kaminska, and G. Quesnel, "An integrated approach for analog circuit testing with minimum number of detected parameters," in *Int. Test Conf.*, 1994, pp. 631–640.

[20] M. Slamani and B. Kaminska, "Multifrequency analysis of faults in analog circuits," *IEEE Design Test of Computers*, pp. 70–80, 1995.

[21] N. B. Hamida, K. Saab, D. Marche, B. Kaminska, and G. Qusnel, "LIMSoft: Automated tool for design and test integration of analog circuits," in *Int. Test Conf.*, 1996, pp. 571–580.

[22] A. Abderrahman, E. Cerny, and B. Kaminska, "Optimization based multifrequency test generation for analog circuits," *J. Electronic Testing*, vol. 9, pp. 59–73, 1996.

[23] ——, "CLP-Based multifrequency test generation for analog circuits," in *VLSI Test Symp.*, 1997, pp. 158–165.

[24] W. M. Mao *et al.*, "Test generation for linear analog circuits," in *Proc. IEEE Custom Integrated Circuits Conf.*, 1995, pp. 521–524.

[25] S. J. Tsai, "Test vector generation for linear analog devices," in *Proc. IEEE Int. Test Conf.*, 1991, pp. 592–597.

[26] H. H. Zheng, A. Balivada, and J. A. Abraham, "A novel test generation approach for parametric faults in linear analog circuits," in *VLSI Test Symp.*, 1996, pp. 470–475.

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

[28] G. Devarayanadurg and M. Soma, "Dynamic test signal design for analog IC's," in *Proc. Int. Conf. Computer Aided Design*, 1995, pp. 627–629.

[29] W. Verhaegen, G. V. der Plas, and G. Gielen, "Automated test pattern generation for analog IC's," in *Proc. VLSI Test Symp.*, 1997, pp. 296–301.

[30] S. D. Huynh, S. Kim, M. Soma, and J. Zhang, "Automatic analog test signal generation using multifrequency analysis," *IEEE Trans. Circuits Syst.-II*, vol. 46, pp. 565–576, 1999.

[31] W. M. Lindermeir, H. E. Graeb, and K. J. Antreich, "Design based analog testing by characteristic observation inference," in *Int. Conf. Computer Aided Design*, 1995, pp. 620–626.

[32] C. Y. Pan and K. T. cheng, "Pseudorandom testing for mixed-signal circuits," *IEEE Trans. Computer-Aided Design Integrated Circuits Syst.*, vol. 16, pp. 1173–1185, 1997.

[33] ——, "Test generation for linear time-invariant analog circuits," *IEEE Trans. Circuits Syst.-II*, vol. 46, pp. 554–564, 1999.

[34] P. N. Variyam and A. Chatterjee, "Test generation for comprehensive testing of linear analog circuits using transient response sampling," in *Int. Conf. Computer Aided Design*, 1997, pp. 382–385.

[35] K. J. Antreich, H. E. Graeb, and C. U. Wieser, "Circuit analysis and optimization driven by worst case distances," *IEEE Trans. Computer-Aided Design*, vol. 13, pp. 57–71, 1994.

[36] G. Devarayanadurg, P. Goteti, and M. Soma, "Hierarchy based statistical fault simulation of mixed-signal IC's," in *Int. Test Conf.*, 1996, pp. 521–527.

[37] S. Cherubal and A. Chatterjee, "Parametric fault diagnosis for analog systems using functional mapping," in *Design, Automation, Test in Europe*, 1999.

[38] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Statistics*, vol. 19, no. 1, pp. 1–141.

[39] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Massachusetts: Addison-Wesley, 1989.

[40] D. G. Saab, Y. G. Saab, and J. A. Abraham, "CRIS: A test cultivation program for sequential VLSI circuits," in *Proc. Int. Conf. Computer Aided Design*, 1992, pp. 216–219.

[41] P. N. Variyam and A. Chatterjee, "Specification–driven test design for analog circuits," in *Defect and Fault Tolerance in VLSI Syst.*, 1998, pp. 335–340.

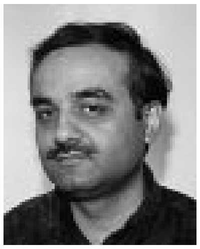
[42] ——, "Enhancing test effectiveness for analog circuits using synthesized measurements," in *VLSI Test Symp.*, 1998, pp. 132–137.

Pramodchandran N. Variyam (S'96–M'99) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Chennai (Madras), India, in 1995. He received the M.S. and Ph.D. degrees in electrical and computer engineering from Georgia Institute of Technology, Atlanta, GA, in 1996 and 1999, respectively.

He joined Texas Instruments, Dallas, TX, in 1999, as a Testability Specialist and worked on improving testability of PLLs and developed test techniques for ADCs. Presently, he is with the Mixed-Signal Wireless Platform product group at Texas Instruments. His research interests include analog and mixed-signal testing, design-for-testability, and test simulation.

Sasikumar Cherubal (S'96–M'02) received the B.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Chennai (Madras), India, and the M.S. degree in electrical engineering from Georgia Institute of Technology, Atlanta, GA. He is currently pursuing the Ph.D. degree at Georgia Institute of Technology.

He is a Principal Engineer at Ardext Technologies. His research interests include the test and diagnosis of analog and mixed-signal electronic circuits and computer-aided design and test automation.



Abhijit Chatterjee (S'86–M'89–SM'94) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1981, the M.S. degree in electrical engineering and computer science from the University of Illinois, Chicago, in 1983, and the Ph.D. degree in electrical and computer engineering from the University of Illinois, Urbana-Champaign, in 1990.

He was with the General Electric Corporate Research and Development Center, Schenectady, NY, from 1983 to 1985 and 1989 to 1992. Since 1993, he has been with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, where he is currently an Associate Professor. He is a principle cofounder of Ardext Technologies, Inc., an Atlanta-based startup providing mixed-signal test services and software, where he is currently Chairman. He is also directing a major program at Georgia Tech in mixed-signal design and tests. He has authored or coauthored over 110 papers in refereed journals and conferences and serves on the program committees of several conferences. His research interests include computer algorithms, reliable design and test of analog and mixed-signal integrated circuits/systems-on-packages/printed wiring boards, and design of low-power integrated circuits and systems.

Dr. Chatterjee received the National Science Foundation (NFS) Research Initiation Award in 1993 and the NSF CAREER Award in 1995. He received two Best Paper Awards from the International Conference on Computer Design in 1992 and VLSI Design in 1993 and three Best Paper nominations. In 1996, he received the Outstanding Faculty for Research Award from the Georgia Tech Packaging.