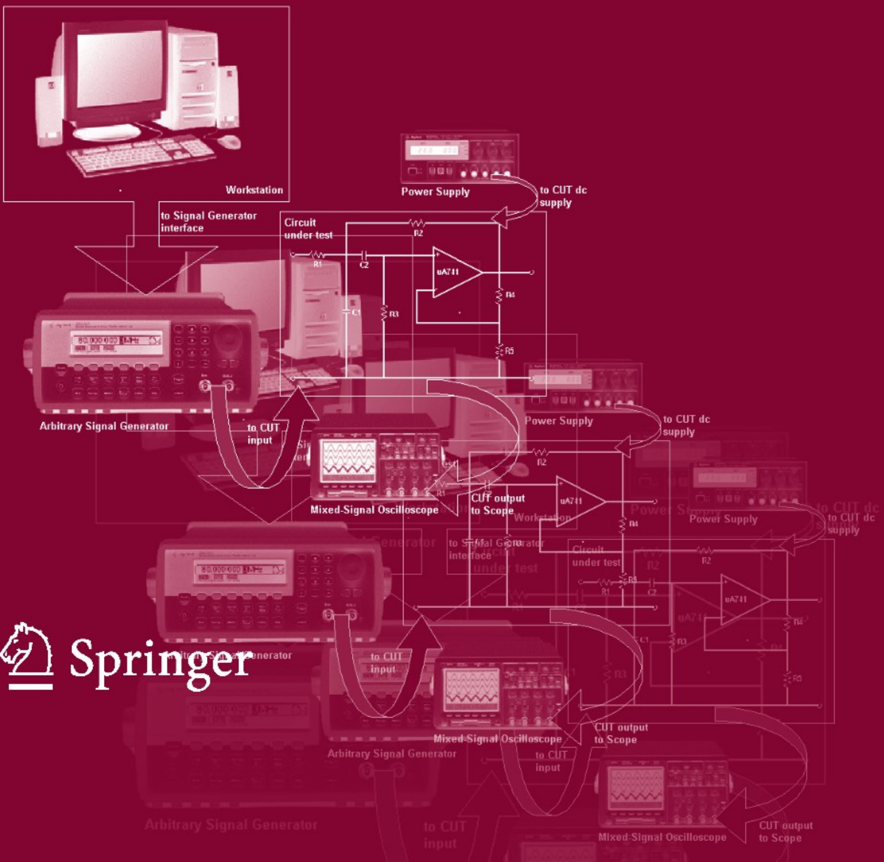
Fault Diagnosis of Analog Integrated Circuits

by

Prithviraj Kabisatpathy, Alok Barua
and Satyabroto Sinha



Springer

FAULT DIAGNOSIS OF ANALOG INTEGRATED CIRCUITS

FAULT DIAGNOSIS OF ANALOG INTEGRATED CIRCUITS

by

PRITHVIRAJ KABISATPATHY
Biju Patnaik University of Technology,
Bhubaneswar, India

ALOK BARUA
Indian Institute of Technology,
Kharagpur, India

and

SATYABROTO SINHA
Indian Institute of Technology,
Kharagpur, India

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN-10 0-387-25742-X (HB)
ISBN-13 978-0-387-25742-6 (HB)
ISBN-10 0-387-25743-8 (e-book)
ISBN-13 978-0-387-25743-3 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springeronline.com

Printed on acid-free paper

Printed in the Netherlands.

Dedication

To our wives

Indrani,

Mausumi

and Gita

To our children

Priyasha,

Arpita,

Santanu and Sandipan

Contents

Preface

The field of integrated circuit has undergone remarkable changes over the past decade. Integrated circuits incorporating both digital and analog function have become more and more popular in semiconductor industry. These combined circuits are called “Mixed Signal Circuits”. Fault diagnosis and testing of integrated circuit has grown into a special field of interest in semiconductor industry. The methodologies for testing of digital circuits are well established. However the methodologies for fault diagnosis of analog integrated circuits are relatively unexplored. Some of the reasons for this under development are lack of proper fault model of the devices operating in the continuous time domain, multiple value of the signal at each node of the circuit, etc.

This book, Fault Diagnosis of Analog Integrated Circuits, is primarily intended as a comprehensive text book at the graduate level, as well as a reference book for practicing engineers in the area of analog integrated circuit testing. Unlike other books, this book does not assume any solid background in analog circuits. A working knowledge of PSpice and MATLAB is necessary. An exposure to artificial neural network (ANN) is desirable.

The prime objective of the book is to provide insight into the different fault diagnosis techniques of analog integrated circuits.

The text is divided into five chapters. Chapter 1 presents an introduction to the basic concepts of fault diagnosis. Both simulation-after-test and simulation-before-test, have been discussed. Fault modelling is presented in Chapter 2. The choice of optimum test stimulus is the key to success in fault diagnosis. All classical and state of the art techniques of test stimulus generation have been discussed in Chapter 3. Chapter 4 presents a feasible fault diagnosis methodology. The state of the art model based observer technique, is discussed in detail. The integrated circuit, particularly the operational amplifier is the most used building block of analog signal processing. This chapter presents a detailed overview of fault diagnosis of operational amplifiers (at the device or component level) using model based observer technique. Simulation results for the fault in bipolar and MOS operational amplifiers are given here. Besides, fault diagnosis of simple analog system-on-chip (SOC) is also presented. An experimental setup for fault detection and diagnosis in analog ICs and SOCs, is also presented in Chapter 4. Chapter 5 rounds off the text with Design-for-test (DfT) techniques and development of Built-in Self-test (BIST) facilities in analog ICs.

The problems at the end of the chapters 2, 3 and 4 may be assigned to the students for work out.

Three appendices present BJT spice parameter values of the μA 741 operational amplifier, MOSFET spice parameter values of the MOS operational amplifier and an introduction to artificial neural network (ANN).

Although an immense effort and attention to detail were exerted to prepare the manuscript, the book may still have some errors due to human nature. The authors welcome suggestions from the readers for improvement of the content and the presentation, for future incorporation.

We take this opportunity to express our appreciation to our numerous students and colleagues whose ideas and suggestions lead to this book. We would also like to acknowledge the support extended by our families; in particular we are grateful to our wives and children for tolerating many absences while we spent hours on this book.

February, 2005
Kharagpur

P. Kabisatpathy
A. Barua
S. Sinha

Chapter 1

INTRODUCTION

System-on-chip (SOC) technology has boosted the importance of analog circuitry, moving it into mainstream integrated circuit (IC) design. ICs with digital and analog circuits on the same substrate are now common [1]. Advances in deep sub-micron technologies have fuelled the explosive increase in IC complexity. The increasing functional complexity and smaller size of these chips makes testing a challenging task, particularly under the constraints of high quality and low price. Considering that testing represents a key cost factor in the production process (a proportion of up to 70% of total product cost is reported in [2]), an optimal test strategy can give substantial competitive edge in the market.

Electronic tests are system dependent and are classified as digital, analog and/or mixed-signal. Current methodologies for the testing of digital circuits are well developed. These include D-Algorithm [3], level sensitive scan design (LSSD) [4], IEEE Standard 1149.1 [5] and built-in logic block observer (BILBO) [6]. By contrast, methodologies for the testing of analog circuits remain relatively underdeveloped due to the complex nature of analog signals. Despite the translation of many analog electronic functions into their digital equivalents, there still exists a need to incorporate analog sections on many chips. In a primarily digital system, analog circuitry is still used to convert speech signals to digital signals, sensor signals are conditioned into microprocessors, digital bit streams are converted to radio frequency (RF) modulation patterns or horizontal scan lines on a computer graphics screen and microprocessors send control signals to actuators. Even with a pure digital system, there still exist power supplies, pull-up resistors to do level shifting and capacitors for electromagnetic compatibility (EMC). All these analog components on a digital system need to be tested during production and field service. In addition, "*when digital clock rates get really high, the 0's and 1's don't have real meaning anymore. The behaviour is essentially analog*" [7]. Therefore, the importance of analog testing cannot be underrated and there is a requirement to develop strategy, which will allow the analog and digital parts of the circuit to be tested simultaneously.

Analog and mixed-signal (AMS) ICs are gaining popularity in applications such as consumer electronics, biomedical equipments, wireless communication, networking, multimedia, automotive process control and real-time control systems. With such wide applications, AMS ICs will constitute the bulk of future electronic devices, making it imperative to research AMS testing, which include digital and analog testing. Compared to digital testing, analog testing lags far behind in both methodologies and tools and therefore demands substantial research and development effort. The objective of this book is to study testing and fault diagnosis of analog and/or the analog portion in AMS ICs. New analog test methodologies need to be compatible with existing digital test methods and be practical in compromising test coverage and test overhead.

1.1 Basic concepts

Historically, digital and analog testing techniques have developed at very different paces, causing analog test methodology to lag behind its digital counterpart. Computer-aided design (CAD) tools for automatic test generation and test circuitry insertion are available since two decades for digital circuits. The main reason for this is the ease of formulating the test generation as a mathematical problem due to the discrete signal and time values. The distinction between what does and what does not work is crisp and clear for digital circuitry. For analog, on the other hand, the question can better be stated as 'how good' the circuitry works. Does an analog-to-digital converter (ADC) work correctly, when it delivers a signal-to-noise ratio (SNR) of 69.99dB, when the specified minimum performance figure is 70dB? Does the system application suffer from this underperformance? For analog designs, the definition of fault-free and faulty circuits is much more a matter of specification thresholds and sensitivity of application than a sharp distinction as in the case of digital circuits.

For analog circuitry, generation of optimal test signals based on design topology is still not fully automated. As opposed to the digital approach based on the gate-level net-list, analog testing still relies mainly on a black-box approach, where the specifications of the circuitry are verified without paying attention to the structure or circuit layout. Another unresolved issue is the usage of standard design methods like the design-for-testability (DfT), to make the circuitry easily or better, testable. Scan chains are used in synchronous digital circuitry for this purpose. A comparable analog approach does not exist. Similarly, modelling of process defects and the use of these models for developing and improving test signals is already standard practice for digital circuits whereas similar methods for analog circuitry are just beginning to be applied by some manufacturers. For some of these issues, alternative approaches are still in research phase. For some others, there are technically feasible alternatives but the existing production infrastructure and cost of changing present test methods is slowing down the acceptance of these methods.

For digital circuits, algorithms for the generation of test patterns based on gate-level net-list exist since as early as 1960's [8][9]. Without the so-called automatic test pattern generation (ATPG) methods, it would have been impossible to produce the large digital ICs of the last twenty years at reasonable costs and quality. A similar test generation solution for analog testing became necessary with the increasing integration of analog and digital functionality on one chip. The analog test community has also been aiming at a solution comparable to that in digital, but the analog version of the problem is not solvable by similar analytical techniques. In the case of digital circuits, the discreteness in time and signal values, the well-defined fault propagation paths and topological boundaries of fault influence have simplified the problem to some extent when compared to the analog case. As a result, algorithms have been developed based on calculating the signal changes introduced by faults and logic rules to find input combinations to create changes between fault-free and faulty behaviour (path sensitisation) and propagate these changes to the primary outputs. A similar approach cannot be applied to analog circuits. The main reasons for this are:

1. There are not only two choices of analog signal values to choose from, but in principle, an infinite number of signal values are possible. The choice between two specific signal values can cause better or less good results related to observing the fault at the outputs.
2. The time variation properties of analog signals bring an extra dimension to the problem, since applying an AC, DC or transient test can be more or less efficient depending on the circuit and the targeted fault.
3. It is not possible to make a one-to-one link between the function and structure of analog circuitry such as in digital circuits. Given a particular topology, there is no general way of determining which part of the functionality is of interest and what the related performance limits are.
4. The propagation of fault effects to the output in analog circuits is not as simple as in the digital case, because of two reasons. First, the effect of a fault cannot be modelled to propagate in one direction, as is the case in digital. The fault effect propagates in all directions and the calculation of this propagation pattern becomes therefore much more complex than in the digital case. Secondly, in analog circuits,

the information that a fault is present at a certain node does not readily comprise the signal value information for that node, making time-consuming calculations of signal values necessary. Nonlinearity, loading between circuit blocks, presence of energy-storing components and parasitic elements further complicate these calculations.

The obstacles presented above have prevented analog net-list based (i.e., structural) test generation from being applied in practice. Unfortunately, a satisfying solution to the analog problem has not been found to this day. The research on this subject [10][11][12] has been going on for decades already, and the results are not good enough such that any analog block design can be made from a net-list description [13]. The alternative of solving the analog test generation problem based on these methods remains unfeasible for the time being and the application of net-list based test generation in practice requires a breakthrough in terms of the computational costs and general applicability. Today, specification based testing (checking whether the specifications are met) and functional testing (checking the functioning of the circuit with a standard input) are the dominating methods in analog testing.

Tests can be performed at several levels including wafer level, package level, module level, and system level or even in the field service level. A commonly mentioned rule of thumb test is the *rule of ten* which suggests that the cost of detecting a bad component in a manufactured part increases tenfold at each level [14]. Therefore, a fault should be detected as early as possible. Even though it is better to find a fault earlier, some later process changes may introduce additional failures. Therefore, an optimisation of testing effort or some compromise should be incorporated in the test procedures.

Tests can be classified into fault detection, fault location or fault prediction. In the manufacturing process or during maintenance, a quick check is needed to pass the good parts and reject the bad parts for maximum product throughput. So only fault detection is needed to evidence the faults. At other times, fault location is needed to detect failed modules or components for repair. Fault prediction is used mainly with highly reliable products or safety related products. Fault prediction continuously monitors the circuit under test (CUT) to identify whether any of its elements are about to fail allowing for a preventive repair. The choice between fault location and fault detection calls for a compromise. Fault location needs better isolation of the components and provides better test coverage. It may be used during both production and repair, but it may slow down the testing process and the throughput. The IEEE 1149.1 [5] standard provides decent fault isolation in digital circuits. For AMS circuits, the proposed IEEE P1149.4

standard [15] is still not widely used. In order to locate every fault, most nodes need to be accessed in the circuit, a requirement that may be impossible to meet in modern fabrication process.

The current approach to detect manufacturing faults in electronic circuits uses several forms of automatic test equipment (ATE), in-circuit tester (ICT) and functional tester. After assembly, the in-circuit testing is done to check for part orientation, part value, part type, and correct connections between all parts. Software models of each part are used to generate the test patterns used by the ICT. The main assumption is that in-circuit testers require physical access to nodes or points on the circuit in order to perform the necessary testing. A functional tester is usually customised to a particular product. The test is performed after packaging the product to check the functionality of the product. Due to the shrinkage in dimension of the electronic components and circuits, it is impossible to access every test point by an ATE. Therefore, alternative methods must be found to address these challenges. IEEE standard 1149.1 [5] and IEEE proposed standard P1149.4 [15] are such alternatives, which provide virtual probes.

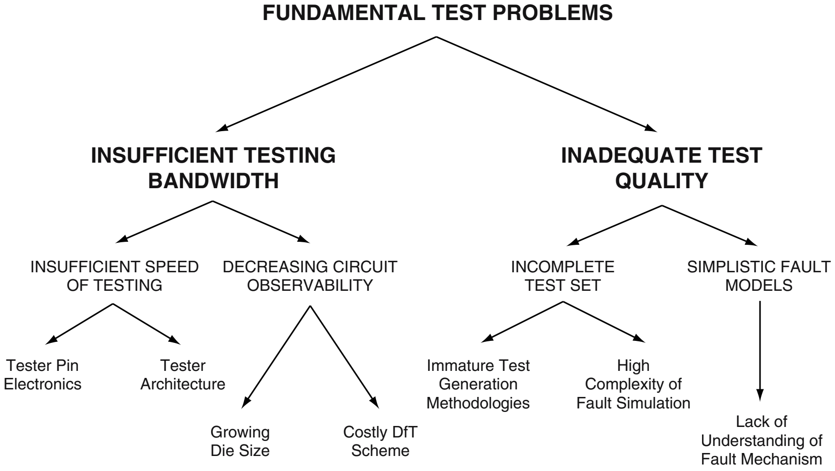

FUNDAMENTAL TEST PROBLEMS
INSUFFICIENT TESTING BANDWIDTH
INADEQUATE TEST QUALITY
INSUFFICIENT SPEED OF TESTING
DECREASING CIRCUIT OBSERVABILITY
INCOMPLETE TEST SET
SIMPLISTIC FAULT MODELS
Tester Pin Electronics
Tester Architecture
Growing Die Size
Costly DfT Scheme
Immature Test Generation Methodologies
High Complexity of Fault Simulation
Lack of Understanding of Fault Mechanism

Figure 1.1: The fundamental test problems and their causes [16].

In Figure 1.1, an analysis of the fundamental problems in testing is depicted by Maly [16]. In this diagram, the two fundamental problems are identified as test bandwidth and test quality and specific challenges are pointed out for solving these problems. For mixed-signal testing specifically, other test hardware-related challenges such as noise level and accuracy of timing-measurements will have to be added to this picture. To summarise, the important issue here can be defined as developing the concept of a framework linking analog and mixed-signal IC design and test environment. This framework must enable an IC development team to

- develop test programs in shorter time,
- debug test programs in shorter time,
- automatically generate test programs that will guarantee high IC quality.

1.2 Historical Background

Algorithms aimed at diagnosing component failures on printed circuit board (PCB) can be applied to identify faulty components in analog ICs. Given limited accessibility to internal nodes in analog ICs, a large number of algorithms and theoretical findings for fault diagnosis and test signal selection have been developed during the past three decades [17]. Duhamel and Rault [18], and Bandler and Salama [19] have presented two outstanding reviews of analog fault diagnosis. Duhamel and Rault [18] reviewed and assessed the techniques available before 1979 for automatic test generation of analog systems by classifying different methods and providing an extensive bibliography. Bandler and Salama [19] summarised various fault location techniques and algorithms developed before 1985 and added most algorithms developed after 1979. They discussed the fault dictionary approach, the parameter identification approach, the fault verification approach, and the approximation approach. Two excellent books have been published recently covering representative methodologies in analog fault diagnosis and modern approaches to mixed-signal testing [20][21].

The measures of testability and the degree of algorithm complexity are the basic theoretical topics for fault diagnosis. The testability studies tell whether a CUT is testable or not, for a given methodology, while the degree of complexity tells the effectiveness of the proposed algorithm. Both are related to the specific algorithms and the ways the test equations are formulated. Their effectiveness relates to the kind of faults being targeted. The testability measures can be defined in DC, frequency domain, and time domain.

Analog fault diagnosis methods are generally classified into simulation-after-test (SAT) and simulation-before-test (SBT) [18][19]. SAT methods focus on parameter identification and fault verification and they are very efficient for soft fault diagnosis because they are based on linear network models. However, the major problem in parameter identification is the ability to access test points. Very often, there are not enough test points to test all components or each added test point is too expensive to accept. As an alternative, the fault verification method addresses the problem with limited number of measurements, by which not all parameters of the circuit can be identified at a time. The method assumes that only a few components are faulty and the rest of the network components are within design tolerances.

Checking the consistency of certain network equations identifies faulty components. The ability to test multiple faults is limited by large number of choices of faulty components, which result in combinatorial explosion for large design. The SAT approaches have the disadvantage of high on-line computational complexity, inability to deal with catastrophic faults, error proneness to component tolerances, and high numerical sensitivity. To compromise test coverage and test simulation, SBT methods emphasise on building a fault dictionary in which the nominal circuit behaviours in DC, frequency or time domain are stored. In the test stage, the measured circuit behaviour is compared with the nominal case and the faults are diagnosed. In manufacturing testing, a DC test is reliable and effective. However, when higher test coverage is needed, a frequency test or time domain test provides more information about the circuit under test without adding test nodes. In the following sections the SAT and SBT methods are discussed.

1.2.1 Simulation-after-test methods

Two most representative methods in SAT namely parameter identification and fault verification are discussed. The first task of parameter identification technique is to formulate sufficient number of independent equations from the measurements to determine all component values. A component value that lies outside the design tolerance range specification is identified as a faulty component.

Berkowitz [22] introduced the concept of network-element-value solvability by presenting the necessary conditions for passive networks in 1962. Even without an algorithm, Berkowitz's studies heralded a new research area, analog fault diagnosis. Saeks et al. [23] proposed a method to determine parameter values using voltage and current measurements when a single excitation is applied. Multiple current excitations are applied to a network and voltage measurements are used to identify network parameters by Biernacki and Bandler [24]. Biernacki and Starzyk [25] gave test conditions, which are sufficient conditions for network solvability problems. Trick et al. [26] researched the necessary and sufficient test conditions for a single test frequency and introduced the adjoint circuit concept into fault diagnosis. In this study, the branch voltages of the unknown components should be available so that the component values can easily be found by a linear method. Navid and Willson [27] have given the necessary and sufficient conditions for the element-value solvability of a linear resistive network. They show that one can determine if it is possible to compute the element values from the test terminals by considering only the circuit topology. Ozawa and Kajitani [28] studied the problem of diagnosability of linear active networks using a voltage and current graph. Starzyk et al. [29] used Coates graph and presented topological and graph-theoretic conditions to determine the required number of excitations and voltage measurements for evaluation of faulty elements within a sub network. Ozawa et al. [30] researched generalised Y-Δ transformation with a voltage controlled current source and its application to element-value solvability problems. The inaccessible nodes in a network are eliminated one by one by the transformation and a sequence of networks is obtained. They researched the backward process of Y-Δ transformation to determine the solvability problem. Visvanathan and Sangiovanni-Vincentelli [31] developed a theory for the diagnosability of non-linear circuits with DC inputs. They derived conditions for the local diagnosability and showed that for diagnosable systems, it is possible to obtain a finite number of test inputs that are sufficient to diagnose the system.

All the methods mentioned above deal with DC domain or single frequency excitation. The multi-frequency techniques include research on the test point selection and test frequency selection. Sen and Saeks [32] introduced a testability measure via multi-frequency measurements for linear systems and its applications to test point selection. Sen and Saeks [33], and Chen and Saeks [34] further discussed and refined this approach. Sen and Saeks' measure is appealing because it provides a quantitative measure of testability and it leads to an efficient computational algorithm. Priester and Clary [35] proposed a testability measure based on optimal experiment designs borrowed from system identification theory. Their measure provides more information on the degree of difficulties about the testability. Rapisarda and Decarlo [36] proposed the tableau approach with multi-frequency excitation for analog fault diagnosis instead of transfer function oriented algorithms. Abderrahman et al. [37] used optimisation techniques to generate multi-frequency test sets for parametric and catastrophic failures. Sheu and Chang [38] proposed an efficient frequency domain relaxation pseudo-circuit approach and the associated solvability conditions with reduced dimension and practical implementation scheme.

The time domain approach includes formulating testable equations, which are solvable from time domain measurements. Saeks et al [39] published an excellent work on dynamic testing. They extended their diagnosability theory for linear systems [33] and memory-less non-linear systems [31] by presenting a necessary and sufficient condition for the local diagnosability of non-linear dynamical systems. Based on a discrete-time circuit description, Visvanathan and Sangiovanni-Vincentelli [40] derived a necessary and sufficient condition for the local diagnosability of a class of non-linear dynamical circuits whose branch relations are analytic functions of their argument. Dai and Souders [41] proposed an efficient approach for functional testing and parameter estimation of analog circuits in time domain based on a sensitivity matrix. Salama and Amer [42] developed a technique based on identifying the discrete time transfer function coefficients of the circuit under test from time domain response. Walker et al. [43] developed a two stage SAT fault diagnosis technique based on bias modulation. The first stage, which diagnoses and isolates faulty network nodes, resembles the node fault location method [44]. The second stage, a sub-network branch diagnosis extracts faulty network parameters. The branch diagnosis is achieved by element modulation, a technique to vary the value of the element externally as a modulated element.

Other studies on parametric identification methods include, parametric testing by artificial neural network processing of transient responses [45], parametric fault diagnosis using functional mapping [46], methods using wavelet transform of the response signal of the device under test and non-deterministic parametric fault model [47], the aggregation of the digitised output values for detecting parametric faults in linear time-invariant circuits [48].

The works discussed so far focussed on the parameter identification method. The fault verification methods use almost the same equations as are used in the parameter identification approaches, except that in the fault verification approaches, circuit components are partitioned into two classes, a fault-free class (class1) and a faulty class (class2). It is assumed that all components in class1 are fault-free and all faults are localised in class2. Using the measurement data and the nominal characteristics of all circuit components, test equations are formulated and expressed as functions of deviations of class2 components. Test equations are over determined and can be satisfied only if all faults are indeed localised in class2. This technique of making assumptions on faults and checking their validity is called *fault verification*.

Another quantitative testability measure, which is based on sensitivity was introduced by Temes [49] and was further studied by Dejka [50]. Skala et al. [51] studied a sensitivity algorithm for checking the consistency or inconsistency, of certain linear equations which are invariant on fault elements. Dai and Souders [41] proposed a testability measure using time domain sensitivity equations. Slamani and Kaminska [52] used sensitivity equations and optimisation method for selecting test sets.

From the topology point of view, Huang et al. [44] introduced k-node-fault testability in 1983. In their studies, node-voltages and nodal equations are used. The beauty of their testability condition is that it depends only on the graph of the circuit instead of the component values. Lin et al. [53] studied the topological conditions for single-branch- fault. Maeda et al. [54]

presented necessary and sufficient analytical as well as graph theoretic conditions for fault detectability and distinguishability in non-linear systems and used these conditions to derive an algorithm for fault diagnosis. Ozawa et al. [55], and Starzyk and Dai [56] presented a decomposition approach for testing large analog non-linear networks. This was found to be superior to conventional methods using sensitivity approach. Recent studies based on the k-fault diagnosis method deal with the problem of component tolerances using artificial backward propagation neural network [57][58]. Other prominent works reported are for unambiguous fault diagnosis based on ambiguity test algorithms [59][60][61][62], selection of minimal set of test nodes [63][64], parametric fault modelling [65] and parallel analog functional test [66].

No matter what kind of testability measures are used, whether it is based on frequency domain, time domain or topology, the advantage is that the measure tells whether the CUT is testable or diagnosable. However, the computational complexity is a difficult problem to overcome. In manufacturing testing, this problem becomes more severe. SBT methods provide a compromise by shifting the computational burden to simulation before test.

1.2.2 Simulation-before-test methods

Martens and Dyck [67] used a frequency domain approach for single element faults. They considered a transfer function as a bilinear function of network elements. Morgan and Towill [68] included the higher order harmonics in the dictionary for the frequency domain response of the network. Varghese et al. [69] utilised the Euclidian norm to normalise the network response deviations. Lin and Elcherif [70] considered DC inputs to build the fault dictionary. Seshu and Waxman [71] employed a frequency domain approach to construct a fault dictionary of a linear frequency-dependent circuit.

Time-domain analysis approaches were also studied to construct the fault dictionary. Macleod [72] used pseudo-random sequences as inputs for analog fault diagnosis. For each faulty condition, the deviation in the impulse response of the network is computed using a periodic pseudo-random sequence as excitation. These deviations are quantised and are stored in the dictionary. Schreiber [73] proposed the test signal design method to construct the fault dictionary. In this, the loci of all single-element drift failure fault signatures are drawn in the augmented signal space to generate the fault dictionary. Wang and Schreiber [74] utilised a complementary signal approach for go/no-go testing of a partitioned network under test. Balivada et al. [75][76] have studied the effects of various stimuli on the variations in delay, rise time, and over shoot that indicate faulty behaviour. Al-Qutayri and Shepherd [77] presented a time-domain go/no-go testing strategy for analog macros. Their strategy is based on exciting an analog macro with a pseudo-random binary sequence and measuring the transient response at the external nodes. With an increasing number of macro-based designs in application, testing macros becomes attractive and necessary.

Different types of measurements were used in the literature to construct the dictionary. The widely used measurements are node voltage [78][70], magnitude and phase of node voltages [79], and voltage/current measurements [80]. Power supply current and voltage measurements are also used by Papakostas and Hatzopoulos [81] who suggested power supply current measurements in linear bipolar ICs for fault detection. They also describe a simple current sensor for on-chip current monitoring, giving accurate analog output proportional to variation of the quiescent current [82]. Somayajula et al. [83] proposed construction of fault dictionaries from the currents in the power supply bus. A ramping power supply is applied at the DC power supply inputs to force the transistors in the circuit to operate in all possible regions. The signatures are then clustered into different groups using a Kohenen neural network classifier. Aain et al. [84] suggested testing of analog circuits by power supply voltage control. The power supply voltage was varied and output voltage was measured for the operational amplifier (Op-Amp) circuits to detect faults, which were otherwise difficult to find with conventional input signal excitation. They also discussed application of AC power supply voltage and compared the fault coverage of voltage level and supply monitoring schemes [85]. In addition, they discussed the effect of changing the power supply frequency on the testing of analog ICs. Lindermeir et al. [86] dealt with the detection of parametric and catastrophic faults by “I/sub DD/monitoring”. ADC quantisation noise is also systematically considered throughout the method. Manhaeve et al. [87] describe the development, evaluation and validation of a structural, cost effective and quantifiable analog and mixed-signal test methodology, applicable in a production test environment and based on the application of supply current testing. An overview of the test suite defining the mix of functional, structured, the quiescent power supply current of the transistor (IDDQ) and built-in self-test (BIST) facilities are offered by Matthes and Ford [88].

Artificial intelligence (AI) and neural network methods have been widely used in analog fault diagnosis, especially in SBT methodologies [89][90][91]. Hatzopoulos and Kontoleon [89] proposed a method, which provides the knowledge base to the computer with qualitative and quantitative, nominal and faulty, element models and their possible failure symptoms.

Manetti et al. [90] proposed methods to generate test point selection using AI techniques, based on a knowledge base constituted by simple rules derived from experience and heuristic reasoning. Based on the back propagation paradigm, several authors studied analog fault diagnosis [80][92][93][94][95]. Rutkowski [80] employed back-propagation network for locating faults in a non-linear dc circuit. Yu et al. [92] proposed the neural network approach for the fault diagnosis of complementary metal-oxide semiconductor (CMOS) Op-Amps with gate oxide short faults. In this study, neural network is trained to respond for the variations in supply current. Wu and Meador [93] suggested a feed forward neural network approach for IC fault diagnosis in a large-scale production-testing environment. However, all these methods present the problem of long training time to converge, even with a relatively small number of training samples. Spina and Upadhyaya [94], and Nissar and Upadhyaya [95] used multi-layer perceptrons as fault classifiers. Yang et al. [96] claimed to have applied a "robust heteroscedastic probabilistic neural network" to analog fault detection and classification. Some other authors [97][98] have used multi-layer perceptrons for diagnosis of multiple faults in analog circuits. Others have used adaptive digital filters in analog fault diagnosis [99][100][101]. Recently a comprehensive examination of neural network architectures for analog fault diagnosis has been reported [102].

Probabilistic and statistical techniques for analog fault diagnosis have been proposed in some of the studies reported. Priester and Clary [35] developed measures of testability and test complexity for linear analog circuit failure analysis based on probabilistic techniques. Favalli et al [103] presented a probabilistic approach to detect analog faults that depends on the conductance of faulty and fault-free networks. They gave methods to find detection probability of each fault and the expected coverage of analog faults. Epstein et al. [104] presented statistical techniques using discrimination analysis and hypothesis testing for fault detection and classification in linear ICs. Devarayanadurg et al. [105] have presented a hierarchy based statistical fault simulation of mixed-signal ICs. They also studied methods for constructing optimal tests to detect structural faults in analog ICs in the presence of process variation by a combination of analytical and numerical optimisation techniques [106]. Gielen et al. [107] and Wang et al. [108] suggested use of *a priori* simulated probability information combined with actual measurement data to decide whether the circuit was faulty or not. Ozev et al. [109] have discussed methodology for testing analog and digital components in a signal path using hierarchical test generation and system level test composition.

The research results discussed provide theoretical basis for analog fault diagnosis. The basic idea of analog fault diagnosis is to formulate equations based on circuit topology and voltage/current measurements so that the desired testability is reached. Unfortunately, most algorithms cannot avoid the computational complexity problem and therefore, they are rarely used in practice. The most practical methods in analog fault diagnosis are simulation and the fault dictionary approach.

A fault dictionary constructs a look up table, which lists each faulty case and corresponding nominal case for comparison purpose. The objectives of fault detection or diagnosis must be clear because they are critical aspects for deciding the fault detection or diagnosis capability of the dictionary. They also have an impact on the size of the dictionary and impose a limitation on the dictionary approach. Too broad fault coverage may end up with prohibitively large number of combinations which may not be realisable algorithmically, while too narrow fault coverage may not meet the quality target. The anticipated faults and the nominal circuit of the CUT need to be simulated in order to develop sets of stimuli and responses to detect and isolate the faults. To generate a reasonable fault list, physical failures and failure modes have to be related and suitable fault models have to be developed. In order to develop fault models for electronic components and devices, failure modes and physical failures are discussed in the next chapter.

1.3 Summary

It is well known fact that ICs with digital and analog circuits having the same substrate are widely used in communication, networking, multimedia, process control, real-time control systems, and biomedical instrumentation. The analog circuits in these mixed-signal ICs serve as interface between the digital processing circuitry and the real-world analog signals. In addition, in the micro-electro-mechanical systems (MEMS), mechanical and electromechanical components are integrated with electronic circuits on a single substrate. It is, therefore, reasonable to expect that in the near future analog and mixed-signal (AMS) circuits will be present in a significant number of mass manufactured application specific integrated circuits (ASICs). As with digital ICs, methods to adequately test AMS circuits on VLSIs are of immense interest to the engineers in manufacturing industry.

The objective is to improve the understanding of testing and fault diagnosis of analog integrated circuits and modules under constraints of manufacturing. These constraints include testing cost and reliability, testing time and accessibility of tested components, as well as IEEE standards and industrial practices. In particular, the topics covered in this book are: testing, fault and fault classification, test stimulus generation, fault detection and identification strategies in analog integrated circuits, design for testability and built-in self-test concepts for analog integrated circuits.

Testing is beneficial for detecting defects and assuring quality, but it has impact on production cost and time-to-market. A favourable solution is to have one methodology, which can be reused in testing chips, modules, and systems. IEEE boundary scan standard provides such methodology but the additional cost of its integration may offset its benefit. The inaccessibility of test nodes and lengthened development lead-time are major concerns and in AMS testing those concerns are heightened. Some of the difficulties in analog testing spawn from the inherent non-linear property of the analog signal, lack of proper fault models, susceptibility to distortion, measurement errors, and node inaccessibility.

References

[1] M. Ismail and T. Fiez, eds., Analog VLSI: Signal and Information Processing, McGraw-Hill Publishing Company, Inc., New York, 1994.

[2] B. Koenemann, “Creature from the deep submicron lagoon”, Invited Lecture, 10th Workshop Test Methods and Reliability of Circuits and Systems, March 1998.

[3] A.N. Airapetian and J.F. McDonald, “Improved test set generation algorithm for combinational logic control”, Digest of Papers, 9th Annual International Symposium on Fault-Tolerant Computing, pp. 133-136, June 1979.

[4] E.B. Eichelberger and T.W. Williams, “A logic design structure for LSI testing”, Proceedings, IEEE/ACM 14th Design Automation Conference (DAC 1977), pp. 462-468, June 1977.

[5] IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE Standard 1149.1 - 1990 (includes IEEE Standard 1149.1a - 1993), The IEEE, Inc., New York, October 1993.

[6] B. Koenemann, J. Mucha and G. Zwiehoff, “Built-in logic block observation techniques”, Proceedings, 1979 IEEE International Test Conference, pp. 37-41, October 1979.

[7] “A D&T roundtable mixed-signal design and test”, IEEE Design & Test of Computers, Vol. 10, pp. 80-86, September 1993.

[8] J.P. Roth, “Diagnosis of automata failures: a calculus and a method”, IBM Journal of Research and Development, Vol. 10, pp. 278-291, July 1966.

[9] P. Goel, “An implicit enumeration algorithm to generate tests for combinational circuits”, IEEE Transactions on Computers, Vol. C-30, pp. 215-222, March 1981.

[10] D.M.W. Leenaerts, “TOPICS: a new hierarchical design tool using an expert system and interval analysis”, Proceedings, 17th European Solid State Circuits Conference (ESSIRC 1991), pp. 37-40, September 1991.

[11] G. Debyser and G. Gielen, “Efficient analog circuit synthesis with simultaneous yield and robustness optimisation”, Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-98), pp. 308-311, November 1998.

[12] M. Krasnicki, R. Phelps, R.A. Rutenbar and L.R. Carley, “MAELSTROM: efficient simulation-based synthesis for custom analog cells”, Proceedings, IEEE/ACM 36th Design Automation Conference (DAC 1999), pp. 945-950, June 1999.

[13] S. Ohr, "Synthesis proves to be holly grail for analog EDA", EE Times at http://www.eetimes.com, June 1999.

[14] T.W. Williams, "Design for testability", Computer Design Aids for VLSI Circuits (P. Antogneti, D. O. Peterson, and H de Man, eds.), NATO AIS Series, Martinus Nijhoff Publishers, The Netherlands, pp. 359-415, 1986.

[15] IEEE Standard for a Mixed-Signal Test Bus (Draft 18), The Proposed IEEE Standard P1149.4, The IEEE, Inc., New York, June 1997.

[16] W. Maly, H.T. Heineken, J. Khare and P.K. Nag, "Design-manufacturing interface: part I - vision", Proceedings, Design Automation and Test in Europe Conference and Exhibition (DATE 1998), pp. 550-556, February 1998.

[17] L.S. Milor, "A tutorial introduction to research on analog and mixed-signal circuit testing", IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing, Vol. 45, pp. 1389-1407, October 1998.

[18] P. Duhamel and J.C. Rault, "Automatic test generation techniques for analog circuits and systems: A review", IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 441-440, July 1979.

[19] J.W. Bandler and A.E. Salama, "Fault diagnosis of analog circuits", Proceedings of the IEEE, Vol. 73, pp. 1279-1325, August 1985.

[20] B. Vinnakota, Analog and Mixed-Signal Test, Prentice-Hall, Inc., New Jersey, 1998.

[21] M. Burns and G.W. Roberts, An Introduction to Mixed-Signal IC Test and Measurements, Oxford University Press, New York, 2001.

[22] R.S. Berkowitz, "Conditions for network-element-value solvability", IRE Transactions on Circuit Theory, Vol. CT-9, pp. 24-29, March 1962.

[23] R. Saeks, S.P. Singh and R-W. Liu, "Fault isolation via component simulation", IEEE Transactions on Circuit Theory, Vol. CT-19, pp. 634-640, November 1972.

[24] R.M. Biernacki and J.W. Bandler, "Postproduction parameter identification of analog circuits", Proceedings, IEEE International Symposium on Circuits and Systems, pp. 1082-1086, April 1980.

[25] R.M. Biernacki and J.A. Starzyk, "Sufficient test conditions for parameter identification of analog circuits based on voltage measurement", Proceedings, European Conference on Circuit Theory and Design, Vol. 2, pp. 233-241, September 1980.

[26] T.N. Trick, W. Mayeda and A.A. Sakla, "Calculation of parameter values from node voltage measurement", IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 466-474, July 1979.

[27] N. Navid and A.N. Willson, Jr., "A theory and an algorithm for analog circuit fault diagnosis", IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 440-456, July 1979.

[28] T. Ozawa and Y. Kajitani, "Diagnosability of linear active network", IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp 485-489, July 1979.

[29] J.A. Starzyk, R.M. Biernacki and J.W. Bandler, "Evaluation of faulty elements within linear subnetworks", International Journal of Circuit Theory and Applications, Vol. 12, pp. 23-27, January 1984.

[30] T. Ozawa, S. Shinoda and M. Yamada, "An equivalent-circuit transformation and its application to network-element-value calculation", IEEE Transactions on Circuits and Systems, Vol. CAS-30, pp. 432-441, July 1983.

[31] V. Visvanathan and A. Sangiovanni-Vincentalli, "Diagnosability of nonlinear circuits and systems - part I: the dc case", IEEE Transactions on Circuits and Systems, Vol. CAS-28, pp. 1093-1102, November 1981.

[32] N. Sen and R. Saeks, “A measure of testability and its applications to test point selection theory”, Proceedings, 20th Midwest Symposium on Circuits and Systems, pp. 576-583, August 1977.

[33] N. Sen and R. Saeks, “Fault diagnosis for linear systems via multifrequency measurements”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 457-465, July 1979.

[34] H.M.S. Chen and R. Saeks, “A search algorithm for the solution of multifrequency fault diagnosis equations”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 589-594, July 1979.

[35] R.W. Priester and J.B. Clary, “New Measures of testability and test complexity for linear analog failure analysis”, IEEE Transactions on Circuits and Systems, Vol. CAS-28, pp. 1088-1092, November 1981.

[36] L. Rapisarda and R. DeCarlo, “Analog multifrequency fault diagnosis”, IEEE Transactions on Circuits and Systems, Vol. CAS-30, pp. 223-234, April 1983.

[37] A. Abderrahman, E. Cerny and B. Kaminska, “Optimization-based multifrequency test generation for analog circuits”, Journal of Electronic Testing: Theory and Applications, Vol. 9, pp. 59-73, August/October 1996.

[38] H-T. Sheu and Y-H. Chang, “The relaxation pseudocircuit method for analog fault diagnosis”, International Journal of Circuit Theory and Application, Vol. 24, pp. 201-221, March/April 1996.

[39] R. Saeks, A. Sangiovanni-Vincentalli and V. Visvanathan, “Diagnosability of nonlinear circuits and systems - part II: dynamical systems”, IEEE Transactions on Circuits and Systems, Vol. CAS-28, pp. 1103-1108, November 1981.

[40] V. Visvanathan and A. Sangiovanni-Vincentalli, “A computational approach for the diagnosability of dynamical circuits”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. CAD-3, pp. 165-171, July 1984.

[41] H. Dai and T.M. Souders, “Time-domain testing strategies and fault diagnosis for analog systems”, IEEE Transactions on Instrumentation and Measurement, Vol. 39, pp. 157-162, February 1990.

[42] A.E. Salama and F.Z. Amer, “Parameter identification approach to fault diagnosis of switched capacitor circuits”, IEE Proceedings - G, Circuits, Devices and Systems, Vol. 139, pp. 467-472, August 1992.

[43] A. Walker, W.E. Alexander and P.K. Lala, “Fault diagnosis in analog circuits using element modulation”, IEEE Design & Test of Computers, Vol. 9, pp. 19-29, March 1992.

[44] Z.F. Huang, C-S. Lin and R-W. Liu, “Node-fault diagnosis and a design of testability”, IEEE Transaction on Circuits and Systems, Vol. CAS-30, pp. 257-265, May 1983.

[45] A. Materka and M. Strzelecki, “Parametric testing of mixed-signal circuits by ANN processing of transient responses”, Journal of Electronic Testing: Theory and Applications, Vol. 9, pp. 187-202, August/October 1996.

[46] S. Cherubal and A. Chatterjee, “Parametric fault diagnosis for analog systems using functional mapping 1”, Proceedings, Design, Automation and Test in Europe Conference and Exhibition (DATE 1999), pp. 195-200, March 1999.

[47] M. Pronath, V. Gloeckel and H. Graeb, “A parametric test method for analog components in integrated mixed-signal circuits”, Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-2000), pp. 557-561, November 2000.

[48] Z. Guo and J. Savir, “Algorithm-based fault detection of analog linear time-invariant circuits”, Proceedings, 18th IEEE Instrumentation and Measurement Technology Conference (IMTC 2001), Vol. 1, pp. 49-54, May 2001.

[49] G. Temes, “Efficient methods of fault simulation”, Proceedings, 20th Midwest Symposium on Circuit and Systems, pp. 191-194, August 1977.

[50] W.J. Dejka, “A review of measures of testability for analog systems”, Proceedings, International Automatic Testing Conference (AUTOTESTCON 1977), November 1977.

[51] A.A. Skala, E.I. El-Masry and T.N. Trick, “A sensitivity algorithm for fault detection in analog circuits”, Proceedings, IEEE International Symposium on Circuits and Systems, pp. 1075-1077, April 1980.

[52] M. Slamani and B. Kaminska, “Analog circuit fault diagnosis based on sensitivity computation and functional testing”, IEEE Design & Test of Computers, Vol. 9, pp. 30-39, March 1992.

[53] C-S. Lin, Z.F. Huang and R-W. Liu, “Topological conditions for single-branch-fault”, IEEE Transactions on Circuits and Systems, Vol. CAS-30, pp. 376-381, June 1983.

[54] H. Maeda, Y. Ohta, S. Kodama and S. Takeda, “Fault diagnosis of non-linear systems: graphical approach to detectability, distinguishability and diagnosis algorithm”, International Journal of Circuit Theory and Applications, Vol. 14, pp. 195-209, July 1986.

[55] T. Ozawa, J.W. Bandler and A.E. Salama, “Diagnosability in the decomposition approach for fault location in large analog networks”, IEEE Transactions on Circuits and Systems, Vol. CAS-32, pp. 415-416, April 1985.

[56] J.A. Starzyk and H. Dai, “A decomposition approach for testing large analog networks”, Journal of Electronic Testing: Theory and Applications, Vol. 3, pp. 181-195, October 1992.

[57] D. Ying and H. Yigang, “On the application of artificial neural networks to fault diagnosis in analog circuits with tolerances”, Proceedings, 5th International Conference on Signal Processing (WCCC-ICSP 2000), Vol. 3, pp. 1639-1642, August 2000.

[58] D. Ying, H. Yigang and S. Yichuang, “Fault diagnosis of analog circuits with tolerances using artificial neural networks”, Proceedings, 2000 IEEE Asia-Pacific Conference on Circuits and Systems (APCCAS 2000), pp. 292-295, December 2000.

[59] G.N. Stenbakken and T.M. Souders, “Ambiguity groups and testability”, IEEE Transactions on Instrumentation and Measurement, Vol. 38, pp. 941-947, October 1989.

[60] G.H. Hemink, B.W. Meijer and H.G. Kerkhof, “Testability analysis of analog systems”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 9, pp. 573-583, June 1990.

[61] M. Slamani and B. Kaminska, “Fault observability analysis of analog circuits in frequency domain”, IEEE Transactions on Circuits and Systems, Vol. 43, pp. 134-139, February 1996.

[62] J.A. Starzyk, J. Pang, S. Manetti, M.C. Piccirilli and G. Fedi, “Finding ambiguity groups in low testability analog circuits”, IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, Vol. 47, pp. 1125 - 1137, August 2000.

[63] N.S.C. Babu, “Efficient techniques for fault diagnosis of analog circuits using dictionary approach”, Ph.D. Dissertation, Indian Institute of Technology, Delhi, 1997.

[64] V.C. Prasad and N.S.C. Babu, “Selection of test nodes for analog fault diagnosis in dictionary approach”, IEEE Transactions on Instrumentation and Measurement, Vol. 49, pp. 1289 - 1297, December 2000.

[65] J.A. Soares Augusto and C.F. Beltran Almeida, “Analog fault diagnosis in nonlinear DC circuits with an evolutionary algorithm”, Proceedings, 2000 Congress on Evolutionary Computation (CEC2000), Vol. 1, pp. 609-616, July 2000.

[66] J. Viehland and T. Fairbanks, “Parallel analog functional test”, Proceedings, 2001 IEEE Autotestcon (IEEE Systems Readiness Technology Conference), pp. 616-624, August 2001.

[67] G.O. Martens and J.D. Dyck, “Fault identification in electronic circuits with the aid of bilinear transformation”, IEEE Transactions on Reliability, Vol. R-21, pp. 99-104, May 1972.

[68] C. Morgan and D.R. Towill, “Application of the multiharmonic Fourier filter to nonlinear system fault location”, IEEE Transactions on Instrumentation and Measurement, Vol. IM-26, pp. 164-169, June 1977.

[69] K.C. Varghese, J.H. Williams and D.R. Towill, “Simplified ATPG and analog fault location via a clustering and separability technique”, IEEE Transaction on Circuits and Systems, Vol. CAS-26, pp. 496-505, July 1979.

[70] P.M. Lin and Y.S. Elcherif, “Analogue circuits fault dictionary - new approaches and implementation”, International Journal of Circuit Theory and Applications, Vol. 13, pp.149-172, April 1985.

[71] S. Seshu and R. Waxman, “Fault isolation in conventional linear systems - a feasibility study”, IEEE Transactions on Reliability, Vol. R-15, pp. 11-16, March 1966.

[72] M.J. Macleod, “Comparison of methods of parameter estimation using pseudo random binary sequences”, Electronics Letters, Vol. 9, pp. 342-343, July 1973.

[73] H.H. Schreiber, “Fault dictionary based upon stimulus design”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 529-537, September 1979.

[74] F-L. Wang and H.H. Schreiber, “A pragmatic approach to automatic test generation and failure isolation of analog systems”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 584-585, July 1979.

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

[76] A. Balivada, H. Zheng, N. Nagi, A. Chatterjee and J.A. Abraham, “A unified approach for fault simulation of linear mixed signal circuits”, Journal of Electronic Testing: Theory and Applications, Vol. 9, pp. 29-41, August/October 1996.

[77] M.A. Al-Qutayri and P.R. Shepherd, “Go/no-go testing of analogue macros”, IEE Proceedings - G, Circuits, Devices and Systems, Vol. 139, pp. 534-540, August 1992.

[78] W. Hochwald and J.D. Bastian, “A dc approach for analog fault dictionary determination”, IEEE Transactions on Circuits and systems, Vol. CAS-26, pp. 523-529, July 1979.

[79] A. Pahwa and R.A. Rohrer, “Band faults: Efficient approximations to fault bands for the simulation before fault diagnosis of linear circuits”, IEEE Transactions on Circuits and Systems, Vol. CAS-29, pp. 81-88, February 1982.

[80] G. Rutkowski, “A two stage neural network DC fault dictionary”, Proceedings, 1994 IEEE International Symposium on Circuits and Systems (ISCAS’94), Vol. 6, pp. 299-302, June 1994.

[81] D.K. Papakostas and A.A. Hatzopoulos, “Supply current testing in linear bipolar ICs”, Electronics Letters, pp. 128-130, January 1994.

[82] A.A. Hatzopoulos, S. Siskos and T. Laopoulos, "Current conveyor based test structures for mixed-signal circuits", IEE Proceedings, Circuits, Devices and Systems, Vol. 144, pp. 213-217, August 1997.

[83] S.S. Somayajula, E. Sanchez-Sinencio and J.P. Gyvez, "Analog fault diagnosis based on ramping power supply current signature clusters", IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 43, pp. 703-712, October 1996.

[84] A.K.B. Aain, A.H. Bratt and A.P. Dorey, "Testing analog circuits by power supply voltage control", Electronics Letters, Vol. 30, pp. 214-215, February 1994.

[85] A.K.B. Aain, A.H. Bratt and A.P. Dorey, "Testing analog circuits by AC power supply voltage", Proceedings, 9th International Conference on VLSI Design, pp. 238-241, January 1996.

[86] W.M. Lindermeir, T.J. Vogels and H.E. Graeb, "Analog test design with I/sub DD/ measurements for the detection of parametric and catastrophic faults", Proceedings, Design Automation and Test in Europe Conference and Exhibition (DATE 1998), pp. 822-827, February 1998.

[87] H. Manhaeve, J. Verfaillie, B. Straka and J.P. Cornil, "Application of supply current testing to analogue circuits, towards a structural analogue test methodology", Proceedings, European Test Workshop 1999, pp. 51-56, May 1999.

[88] D. Matthes and J. Ford, "Technique for testing a very high speed mixed signal read channel design", Proceedings, 2000 IEEE International Test Conference, pp. 965-970, October 2000.

[89] A.A. Hatzopoulos and J.M. Kontoleon, "A new approach for automatic fault diagnosis in analog circuits", International Journal of Circuit Theory and Applications, Vol. 18, pp. 387-400, 1989.

[90] S. Manetti, M.C. Piccirilli and A. Liberatore, "Automatic test point selection for linear analog network fault diagnosis", Proceedings, IEEE International Symposium on Circuits and Systems, Vol. 1, pp. 25-28, May 1990.

[91] A. Bernieri and G. Betta, "On-line fault detection and diagnosis obtained by implementing neural algorithms on a digital signal processor", IEEE Transactions on Instrumentation and Measurement, Vol. 45, pp. 894-899, October 1996.

[92] S. Yu, B.W. Jervis, K.R. Eckersall, I.M. Bell, A.G. Hall and G.E. Taylor, "Neural network approach to fault diagnosis in CMOS Op-Amps with gate oxide short faults", Electronic letters, Vol. 30, pp. 695-696, April 1994.

[93] A. Wu and J. Meador, "Measurement selection for parametric IC fault diagnosis", Journal of Electronic Testing: Theory and Applications, Vol. 5, pp. 9-18, August 1994.

[94] R. Spina and S.J. Upadhyaya, "Linear circuit fault diagnosis using neuromorphic analyzers", IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 44, pp. 188-196, March 1997.

[95] A.I. Nissar and S.J. Upadhyaya, "Fault diagnosis of mixed signal VLSI systems using artificial neural networks", Proceedings, South-West Symposium on Mixed-Signal Design, pp. 93-98, April 1999.

[96] Z.R. Yang, M. Zwolinski, C.D. Chalk and A.C. Williams, "Applying a robust heteroscedastic probabilistic neural network to analog fault detection and classification", IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. 19, pp. 142-150, January 2000.

[97] Y. Maidon, B.W. Jervis, N. Dutton and S. Lesage, “Diagnosis of multifaults in analog circuits using multilayer perceptrons”, IEE Proceedings, Circuits, Devices and Systems, Vol. 144, pp. 149-154, June 1997.

[98] M.A. El-Gamal and M.F. Abu El-Yazeed, “A combined clustering and neural network approach for analog multiple hard fault classification”, Journal of Electronic Testing: Theory and Application, Vol. 14, pp. 207-217, June 1999.

[99] L. Carro and M. Negreivos, “Efficient test methodology based on adaptive algorithm”, Proceeding, 35th Design Automation Conference (DAC98), pp. 230-233, June 1998.

[100] E.F. Cota, M. Negreivos, L. Carro and M. Lubaszewaski, “A new adaptive analog test and diagnosis system”, IEEE Transactions on Instrumentation and Measurement, Vol. 49, pp. 223-227, April 2000.

[101] Y-T. Chen and C. Su, “Test waveform shaping in mixed signal test bus by pre-equalization”, Proceedings, 19th IEEE VLSI Test Symposium (VTS 2001), pp. 260-265, April-May 2001.

[102] M. Aminian and F. Aminian, “A comprehensive examination of neural network architectures for analog fault diagnosis”, Proceedings, INNS-IEEE International Joint Conference on Neural Networks (IJCNN2001), Vol. 3, pp. 2304-2309, July 2001.

[103] M. Favalli, P. Olivo and B. Ricco, “A probabilistic fault model for analog faults in CMOS circuits”, Proceedings, European Design Automation Conference (EDAC), pp. 85-88, February 1991.

[104] B.R. Epstein, M. Czigler and S.R. Miller, “Fault detection and classification in linear integrated circuits: an application of discrimination analysis and hypothesis testing”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 12, pp. 103-113, January 1993.

[105] G. Devarayanadurg, P. Goteti and M. Soma, “Hierarchy based statistical fault simulation of mixed-signal ICs”, Proceedings, 1996 IEEE International Test Conference, pp. 521-527, October 1996.

[106] G. Devarayanadurg, M. Soma, P. Goteti and S.D. Huynh, “Test set selection for structural faults in analog ICs”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 18, pp. 1026-1039, July 1999.

[107] G. Gielen, Z. Wang and W. Sansen, “Optimal fault detection for analog circuits under manufacturing tolerances”, Electronics Letters, Vol. 32, pp.33-34, January 1996.

[108] Z. Wang, G. Gielen and W. Sansen, “Probabilistic fault detection and the selection of measurements for analog integrated circuits”, IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. 17, pp. 862-872, September 1998.

[109] S. Ozev, I. Bayraktaroglu and A. Orailogiu, “Test synthesis for mixed-signal SOC paths”, Proceedings, Design, Automation and Test in Europe Conference and Exhibition (DATE 2000), pp. 28-33, March 2000.

Chapter 2

FAULT AND FAULT MODELLING

2.1 Introduction

The rise of system-on-chip (SOC) technology has dramatically boosted the importance of analog circuitry, moving it more into mainstream integrated circuit (IC) design. Analog and digital circuits are now being integrated into a SOC. Advances in deep sub-micron technologies has fuelled the explosive increase in IC complexity. The smaller size makes these chips more sensitive to fabrication variations and tolerance accumulations. Thus there is a growing demand for fault tolerance, which can be achieved not only by improving the reliability of the functional units, but also by an efficient fault detection, isolation and accommodation concept. Consequently, testing and fault diagnosis is becoming one of the major cost factors in the overall IC manufacturing expenses. It has been recognised as a valuable means to (a) check system installation and configuration after maintenance activities, (b) ensure correct system functionality at start-up, and (c) avoid masking and accumulation of errors during operation. Availability of a system (or of redundant system components) can be significantly increased, if testing is employed to allow rapid diagnosis after a failure.

In the context of fault diagnosis, a fault is understood as any kind of malfunction in the system that leads to an unacceptable anomaly in the overall system performance. A fault in a system can be very costly in terms of loss of production, equipment damage and economic setback. Faults are developed in a system due to normal wear and tear, design or manufacturing defects or improper operation leading to stress beyond endurable limits. In many cases degradation in the performance of the system is sustained for some duration before it actually “fails”. In many other cases a system continues to operate with a failed component resulting in degraded performance.

The variety of fault modes that can occur may be classified as:

1. Abrupt (sudden) faults, i.e., step-like changes.
2. Incipient (slowly developing) faults, e.g., drift or bias.

Typically, abrupt faults play an important role in safety-critical applications (e.g. in power plants, transportation systems and drug manufacturing systems, etc,), where a system failure have to be detected early enough so that disastrous consequences arising due to failures can be avoided by early system reconfiguration. On the other hand, incipient faults are of major relevance in connection with maintenance problems where early detection of worn out components is required. In this case faults are typically small and not as easy to detect, but the detection time is of minor importance and may therefore be large.

In the study of fault diagnosis the construction of a fault dictionary using fault simulation techniques are widely used for choosing the test strategy. Some methodologies use schematics as the starting point to generate fault lists in fault simulation. For example, when a fault list is generated, every component is either shorted, or opened, shorted to power, shorted to ground in single fault situation, or a large number of different fault combinations are considered in multiple fault cases. The disadvantage of doing so is that it neglects the physical layout information of the circuitry and hence it could generate some unrealistic faults in the lists or a prohibitively large fault list. Therefore very often a single fault assumption is made. Unfortunately such assumption is often invalid. For instance, a single cut line across a PCB generates multiple faults. One problem in schematic based fault generation without layout information is its inaccuracy in fault models. For example, in CMOS short circuit simulation, a short between nodes should be a proper resistor between nodes instead of zero resistance [1], especially in an integrated circuit [2][3]. Therefore, it is advantageous to study testing by relating the system specifications to details of the layout and process. It is necessary to investigate correlations between fault models and physical failures. Based on the probability of the occurrence of physical failures and the fault behaviours caused by such failures, a realistic fault list can be generated and the fault models can then be built. Fantini and Morandi have presented a review on failure modes and mechanisms for very large scale integration (VLSI) ICs [4]. Failure mechanisms for electronic components are presented briefly in the following section.

2.2 Failure modes in electronic components

A failure mode is the effect by which a failure is observed, while a failure mechanism is the chemical, physical, or metallurgical process, which leads to component failure [5]. In electronic components, there exist different failure modes such as open circuit, short circuit, degraded performance and functional failures. Degradation faults depend mainly on variations of certain parameters of the components used in a circuit from its nominal values. This may be due to manufacturing defects, process variations, change in the environment or ambient temperature and/or wear out due to aging. Functional faults, on the other hand, are based on the fact that a circuit may continue to function, but some of its performance specifications may lie outside their acceptable ranges. For example, an amplifier may continue to amplify with a very low gain. Birolini [6] has summarised the relative occurrence of failure modes in some electronic components, which are shown in Table 2-I.

Table 2-I shows that most physical failures are open and short faults, which are comparatively easier to detect than degradation and functional faults. For example, an amplifier may continue to amplify with a very low gain. In printed circuit boards (PCBs), approximately 75% of faults occur at assembly, only 20% are component faults, and 5% are PCB faults [7].

An important application of Table 2-I is to calculate test coverage and yield estimation. For instance, if the open fault for a fixed resistor can be detected, 90% test coverage for that resistor is estimated. Further, faults in electronic components can be classified into following three groups with respect to their origin [8]:

1. *Electrical stress failures:* Being an event dependent failure mechanism, electrical stress is a continuous source of device defects over product lifetime. It is most often caused by improper handling.
2. *Intrinsic failures:* The group of intrinsic failures subsumes all crystal related defects. Since such defects depend very much on the maturity of the manufacturing process, they most often lead to yield loss or infant mortality, rarely to wear-out effects (gate-oxide wear-out and wear-out due to surface charge effects or ionic contamination have been observed). Type and manifestation of intrinsic failures are determined by technology: Gate-oxide defects are specific to metal-oxide semiconductor (MOS) devices by their nature, while current gain shifts are a typical bipolar device defect manifestation. Performance degradation is a long-term effect of intrinsic failures.
3. *Extrinsic failures:* Extrinsic failures comprise all defects related to interconnection, passivation and packaging. They can be classified into three categories with respect to the time of defect manifestation:

- severe process deficiencies resulting in easy-to-detect errors (e.g., open bonding),
- wear-out effects affecting long term reliability (e.g., moisture-related failures) and
- radiation-related errors continuously occurring over product lifetime.

Table 2-I: Relative occurrence of failure modes in some electronic components [6].

components	short %	open %	degradation %	functional %
digital, bipolar ICs	30	30	10	30
digital MOS ICs	20	10	30	40
linear ICs	30	10	10	50
bipolar transistors	70	20	10	--
field-effect transistors	80	10	10	--
diodes, general purpose	70	30	--	--
diodes, sneer	60	30	10	--
diodes, HF	80	20	--	--
SCRs	20	20	60	--
opto-electronic devices	10	50	40	--
resistors, fixed	90	10	--	--
resistors, variable	60	20	20	--
capacitors, foil	80	10	10	--
capacitors, metal foil	40	60	--	--
capacitors, ceramic	50	40	10	--
capacitors, tantalum, dry	60	20	20	--
capacitors, aluminium, wet	20	10	70	--
coils	10	30	--	60
relays	15	15	--	70
crystals	--	80	20	--

The probability of wear-out defects is strongly influenced by the package type. An appreciable percentage of field failures due to packaging can be traced to moisture in the package. The widely used plastic package exhibits the worst quality. Due to their low power dissipation, CMOS devices are more susceptible to corrosion than other devices. The order of importance of the failure mechanisms further depends on parameters like device size, maturity of the technology, and extent and effectiveness of the screening applied after production. With a proportion of 58% [8], electrical stress induced defects play a dominant role in the field failures. A vast majority of failure mechanisms is extremely temperature dependent. High temperature or temperature cycling, lead to significant increase in failure rate, the same applies for high supply voltage also. Table 2-II gives typical examples for each of these groups, for a detailed discussion, please refer to [8].

Table 2-II: ***Global classification of component failures [8].***

failure group	relevant parameters	time distribution of failures
electrical stress	handling	continuous
intrinsic	technology	predominantly infant but also wear-out
	process	yield loss
extrinsic	packaging	wear-out, rarely infant
	radiation	continuous

An effective fault model is a fundamental issue for a successful analog test and diagnosis strategy. In the following section, an attempt has been made to provide an outline of analog fault modelling.

2.3 Analog fault modelling

The fault list is the set of all modelled faults and the test generated by the test process should detect all modelled faults. Realistic analog fault models can be achieved by knowing the behaviour of the circuit. In general, an analog IC under test can have the following three outcomes:

1. *Catastrophic (hard) failure:* The circuit is not functioning at all.
2. *Unacceptable performance degradation:* In this case, the circuit is still functioning, but some of its performance specifications lie outside their acceptable range. Performance degradation is usually referred to as a *soft* failure.
3. *Acceptable performance:* The circuit is functioning and all its performance parameters are within their specification ranges. In this case, the circuit is said to be *correct.*

From above, faults in analog ICs are generally classified in to the following two categories [9]:

- *Catastrophic (hard) faults:* Catastrophic faults are all those changes to the circuit that cause the circuit to fail catastrophically. These faults include shorts, opens or large variations of a design parameter like forward beta (β) in bipolar junction transistors (BJTs) and width and length of MOS field effect transistors (MOSFETs). Catastrophic faults are caused by major structural deformations or extreme out-of-range parameters and lead to failures that manifest themselves in a completely malfunctioning circuit. Electro-migration and particle contamination phenomena occurring in the conducting and metallisation layers are the major causes of opens and bridging shorts.

- *Parametric (soft) faults:* Parametric faults are those changes that cause performance degradation of the circuit. These faults are due to the process fluctuations. These faults involve parameters' deviations from their nominal value that can consequently quit their tolerance band. Parametric faults are due to out-of-specification parameter deviations and so depend on the acceptability band defined by tolerances of process parameters.

As analog faults are continuous in mode they can take an infinite number of values and so, the only difference between catastrophic and parametric faults depend on the concept of "totally malfunctioning circuit". In addition, faults considered catastrophic at one description level may become parametric at a higher one. Further, a good knowledge of the probability of occurrence of all possible defects is necessary for actual fault coverage estimation by a test methodology. From the perspective of physical failure mechanisms, changes to a circuit comes from a wide range of sources [10] such as manufacturing defects, process variations, circuit and environmental parasitic, changes in the environment or ambient temperature, and design errors/non-robustness.

Numerous techniques based on the catastrophic and/or parametric fault models with emphasis on either stimuli design (i.e., test generation) [11][12][13][14][15][16][17][18][19], or response analysis (i.e., signature analysis for deriving the acceptance region) [20][21][22][23] has been proposed. The works in [14][17][18][19] address the test generation problem in the frequency domain for linear time-invariant (LTI) circuits. Sinusoidal stimuli with specific frequency, which maximises the output difference between the faulty and fault-free circuit are selected as the input stimuli. Specifically parametric deviation that could possibly mask the faults is considered in the selection of frequency [19]. A symbol based approach for deriving the test frequency is demonstrated in [17]. Methods of test generation in time domain are addressed in [12][13][15][16] which derive static or dynamic time domain test stimuli. Elsewhere, test generation is formulated as a linear [15], quadratic [13] or dynamic [16] programming problem to maximise the output difference. These time domain test generation techniques usually can be applied to both linear and non-linear circuits. Techniques using the tolerance range of device parameters for analog circuit fault detection are proposed in [20][21][22]. Fault-model based techniques for AMS testing of [24][25][26] includes DC voltage/current and pseudorandom excitation techniques. In the DC technique, the DC output voltages/currents are used as signatures. Modelling of analog and mixed-signal ICs for testing and fault diagnosis using standard test stimuli generated for digital ICs have been reported in [27][28][29].

In the above studies, the faults are modelled mostly as open, short, and variable component values. However, component value changes are usually significant in these failure modes. As a result, a faulty value with a value ten times larger or ten times smaller is a reasonable assumption in generating the fault list. Open and short faults are only the extreme cases of these two. Therefore, if ten times larger or ten times smaller faults can be covered, the open and short faults can be detected also. In IC models, short and open should be considered as resistive values according to the technology and process [30]. The information provided in the literatures can be used for making test decisions, creating fault models, generating fault lists, and calculating fault coverage in fault simulation. A comprehensive structured approach for testing and fault diagnosis of AMS circuits and systems have not yet materialised.

The basic problem with analog IC fault diagnosis is the absence of efficient fault models [31], component tolerances and non-linearities. It is difficult to arrive at a general fault model like the stuck-at models for the digital circuits. As described above faults in analog ICs can be classified into two categories:

catastrophic faults or hard faults

parametric faults or soft faults

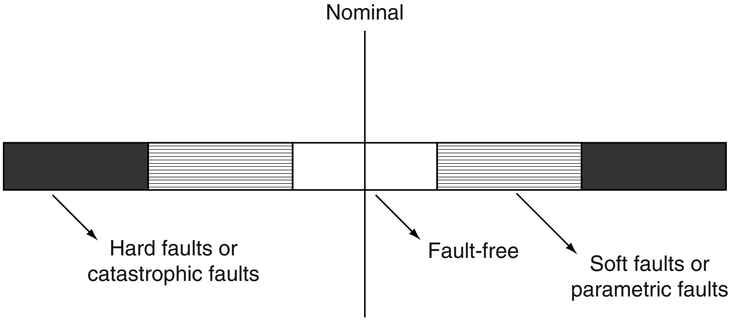

Nominal
Hard faults or catastrophic faults
Fault-free
Soft faults or parametric faults

Figure 2.1: ***Taxonomy of faults.***

Therefore the taxonomy of analog faults can be represented as shown in Figure 2.1. There is a region of acceptable behaviour around nominal range. Beyond this region, there is circuit performance that does not meet design specification, but does not cause complete circuit failure. Finally there are faults that render the circuit inoperable. Since both hard faults and soft faults can take on infinitely many varieties, there are infinitely many analog faults. Consequently, we must choose a subset of faults, which will lead to the best possible fault list.

Since 80-to-90 percent of analog faults involve shorted and open resistors, capacitors, diodes and transistors [6], in this study we have chosen the fault models of various devices as shown in Figure 2.2. Open faults are hard faults in which the component terminals are out of contact with the rest of the circuit creating a high resistance at the incidence of fault in the circuit. Addition of a high resistance in series (e.g., RS ≥ 1 MΩ) with the component (e.g., resistor, capacitor or diode) can simulate the open faults. Short faults, on the other hand, are a short between terminals of the component (effectively shorting out the component from the circuit). A small resistor in parallel (e.g., RP ≤ 1 Ω) with the component can simulate this type of fault for resistors, capacitors and diodes.

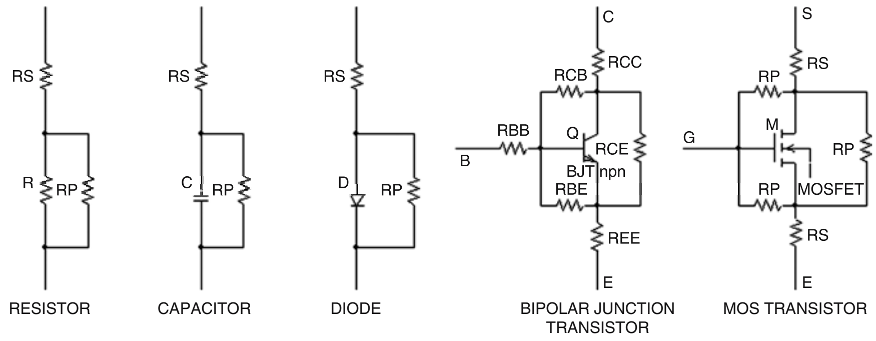


Figure 2.2: ***Fault models of resistor, capacitor, diode and transistor.***

The BJT can have three open faults (at the base, collector and emitter terminals) and three short faults (between base-emitter, collector-base and collector-emitter). These open and short faults are emulated in the same manner using three series resistors RBB, RCC and REE for the open faults and three parallel resistors RBE, RCB and RCE respectively as shown in Figure 2.2. In addition, the BJT has two extreme out-of-range parametric faults for the value of beta (β). A MOSFET has five hard faults: two stuck-open faults at the source and drain, and three stuck-short faults between source-drain, drain-gate and gate-source [32]. These stuck-open faults can be emulated using a high resistance RS in series and the stuck-short faults can be emulated using a small resistance RP between the terminals using the fault model as shown in Figure 2.2.

With this set of fault models we obtain a standard set of faults. The total number of catastrophic faults in a BJT integrated circuit can be

$$N_{CF} = 2(R + C + D) + 8B \tag{2.1}$$

and that in a MOS integrated circuit can be

$$N_{CF} = 2(R + C + D) + 5M \tag{2.2}$$

where R = number of resistors, C = number of capacitors, D = number of diodes, B = number of BJTs, and M = number of MOSFETs.

In practical circuits, the soft faults are the most difficult to model and test. The first problem in the testing of soft faults in analog ICs is to decide on what kind of circuit component deviations from nominal should be considered faulty. Deciding on the tolerance is a major hurdle. The enormous fault list is rendered manageable by quantising the possible values that a circuit under fault can take. The fault free category includes all the values lying in the range N ± σ, where N is the nominal value and σ is the standard deviation from the nominal value. A circuit component having any value in this range is considered fault-free. If it maps to some value outside this nominal range, it is considered faulty. Moreover it could be faulty either above or below the tolerance. While simulating the bipolar analog ICs for fault conditions, these soft faults can be modelled using a reduced set of SPICE parameters like forward and reverse β, junction capacitance, transport saturation current, forward Early voltage, forward and reverse transit time, etc. of the devices. Similarly, for simulating the MOS analog ICs for fault conditions, these soft faults can be modelled using parameters like channel length, channel width, saturation current, threshold voltage, oxide thickness, etc. of the devices.

2.4 Approximation modelling of analog integrated circuits

As described later in Chapter 4, one of the important concepts of fault diagnosis in analog ICs is the use of model-based observer scheme. The development of approximation models based on the physical information and data of the analog IC under test is the foremost goal of such a fault diagnosis methodology. Approximation models that can be used for modelling of failures in any dynamical system may be any one of the various types of mathematical models available in the literature [33]. Some important types of approximation models are given below:

1. *Polynomials:* Polynomial approximation is the most extensively studied approximation method. The class of polynomial functions of degree *n* is given by

$$\hat{f}_n(z;\hat{\theta}) := \left\{ \sum_{i=0}^{n} \hat{\theta}_i z^i : \hat{\theta}_i \in R, \hat{\theta}_n \neq 0 \right\} \tag{2.3}$$

   Polynomials are linearly parameterised approximators and according to the well-known Weierstrass theorem [33], for any function $f \in C[D]$ and any $\varepsilon > 0$, there exist a polynomial $p \in \hat{f}_n$ (for *n* arbitrarily large) such that $\sup_{z \in D} | f(z) - p(z) | \leq \varepsilon$. In the special case of *n=1*, the polynomial expansion reduces to a linear system, which constitutes the best-developed part of system theory.
2. *Rational functions:* Another type of approximation method is the rational function approximation. In this case

$$\hat{f}_{n,m}(z;\hat{\theta},\hat{\vartheta}) := \left\{ \frac{\sum_{i=0}^{n} \hat{\theta}_i z^i}{\sum_{i=0}^{m} \hat{\vartheta}_i z^i} : \hat{\theta}_i, \hat{\vartheta}_i \in R \right\} \tag{2.4}$$

   with the restriction that the zeros of the denominator polynomial are outside the approximation region. In general, rational functions have greater approximation power than polynomial functions, in the sense that with the same number of parameters one is able to obtain better approximation accuracy [33]. Rational functions are nonlinearly parameterised approximators.
3. *Spline functions:* Spline functions are examples of piecewise polynomial approximators [34]. The main idea behind spline functions is the partition of the approximation region into a finite number of sub-regions via the use of *knots.* In each sub-region a polynomial of degree at most *n* is used, with the additional requirement that the overall function is (*n-1*) times differentiable. The most popular type of spline functions is *cubic splines* where *n=3*, i.e., cubic polynomial pieces that are joined so that the overall function is twice differentiable. Spline functions with fixed knots are

linearly parameterised approximators; however, spline functions with variable knots become nonlinearly parameterised approximators.

4. *Artificial neural networks (ANNs):* ANNs are approximation methods based on models of biological signal activity [35]. Although various ANN models have been proposed, by far the most popular is the class of multi-layer ANNs with sigmoid-type activation function. In the case of a two layer ANN

$$\hat{f}_n(z;\hat{\theta},\hat{\vartheta},\hat{\varphi}) := \left\{ \sum_{i=1}^{n} \hat{\theta}_i \sigma(\hat{\vartheta}_i z + \hat{\varphi}_i) : \hat{\theta}_i, \hat{\vartheta}_i, \hat{\varphi}_i \in R \right\} \tag{2.5}$$

where, $\sigma(\bullet)$ is the sigmoid activation function. Theoretical works by several researchers have shown that such networks can uniformly approximate any function $f \in C[D]$ to any degree of accuracy (universal approximation), provided n is sufficiently large, or equivalently the network has sufficiently large number of neurons. Multi-layer ANNs are nonlinearly parameterised approximators.

5. *Radial-bias-function networks:* Another class of neural networks that has attracted considerable attention is the radial-bias-function (RBF) network model. The output of the RBF network is of the form

$$\hat{f}_n(z;\hat{\theta}) := \left\{ \sum_{i=1}^{n} \hat{\theta}_i \omega_i(z) : \hat{\theta}_i \in R \right\} \tag{2.6}$$

where, ω_i is the output of the ith basis function. The Gaussian function $\omega_i(z) := \exp(-|z - c_i|^2 / \sigma_i^2)$, where c_i and σ_i are the ith centre and width respectively, is usually chosen as the basis function. RBF networks are also capable of universal approximation. In many respects, the approximation properties of RBF networks are similar to those of spline functions. For example, if the centre and width are kept fixed then the RBF networks are linearly parameterised approximators; if they are allowed to vary then RBF networks become nonlinearly parameterised approximators.

6. *Adaptive fuzzy systems:* The fuzzy logic paradigm [35] provides another type of approximator. Fuzzy systems approximate functions by covering their graphs with fuzzy patches or fuzzy rules of the form “if antecedent conditions hold, then consequent conditions hold”. The approximation increases in accuracy as the fuzzy patches increase in number and decrease in size. In adaptive fuzzy systems

each fuzzy rule is weighted by adjustable parameters or weights. Fuzzy systems offer the possibility of using linguistic information, based for example on common sense or experts' knowledge, for control of systems where a mathematical model is hard to determine.

To simplify the notations in the above approximation methods, the case of single-variable functions (z is a scalar) are considered. The above list of approximators, although not complete, includes many of the approximation methods used for modelling of dynamical systems. The first three of the approximators described above are based on classical approximation methods while the rest have been proposed in the context of "intelligent control".

The preceding discussion indicates that ANNs represent one class of on-line approximators. Investigations comparing neural networks with other approximation models are still at a preliminary stage; however, from a system engineering perspective, ANNs possess several properties that make them appropriate for approximation of unknown systems, in the context of fault diagnosis. Some of these properties are:

- Massive parallelism,
- Fault tolerance,
- Possibility of analog hardware implementation,
- Convenient adaptation capabilities and
- Good generalisation features.

Furthermore, ANNs have received a great deal of research interest in the past several years and two tutorial articles [36][37] describe the various types of ANNs that are mostly used. ANNs lead to solutions of problems in pattern recognition, associative memory, database retrieval and process fault detection and identification even in the following environment:

- Poorly defined models
- Noisy inputs
- Nonlinear systems

The choice of network architecture is growing. Hsu et al. [38] outlines a comparison of five different ANN architectures and have shown that the backward error propagation (BEP) algorithm provides the best results for the pattern classification task. Many other workers have also had success using the BEP network [39][40][41][42][43][44][45]. These researchers and others provide sufficient confidence in the advantages of the use of ANNs in fault diagnosis of analog ICs. An overview of the ANN is presented in Appendix A.

2.5 Summary

This chapter has presented fault and fault modelling in general and that of analog IC in particular. Section 2.1 gave an introduction to the subject. The definition of the faults has been discussed in this section. This section also presented the fault mechanism.

Section 2.2 discussed the failure modes in electronic components. A clear understanding of failure modes is important to avoid any confusion in the development of analog fault modelling.

Analog fault modelling was presented in section 2.3. Both the catastrophic and parametric faults of BJT and MOSFET have been considered. This section also discussed the different techniques of fault modelling. Fault model of resistor, capacitor, diode and transistors are covered here.

Section 2.4 discussed some of the important approximation modelling of the analog integrated circuits.

Before beginning the study of the following chapters the reader may wish to study the appendix A that presents an overview of the ANN. It covers model of an artificial neuron, the activation function and the structure of an ANN. A good understanding of the ANN will provide a firm foundation for later chapters and subject materials.

Exercises

2.1. What are the different types of failure possible in an electronic component? How do the layout of an integrated circuit and its packages affect failure of the device? Why does the CMOS short circuit simulation use a resistor instead of zero resistance?

2.2. Count the possible number of catastrophic faults for the circuit of Fig. 2.3

2.3. Show that the interest compounded annually over a particular number of years on a principal could be expressed as a polynomial of the interest rate per annum, r. The cumulative increase in the principal, R, could be expressed with an r^0 component included.

2.4. The following data has to be fitted in with a quartic polynomial where the augmented principal is a polynomial function of the interest rate:

Interest rate per annum (r)	3	4	5	6	7	8	9	10
Augmented Principal (R)	108	113	122	126	131	136	141	146

Some uncertainty is added to the data because of the rounding off of R. Use least squares (LS) method to get the polynomial coefficients.

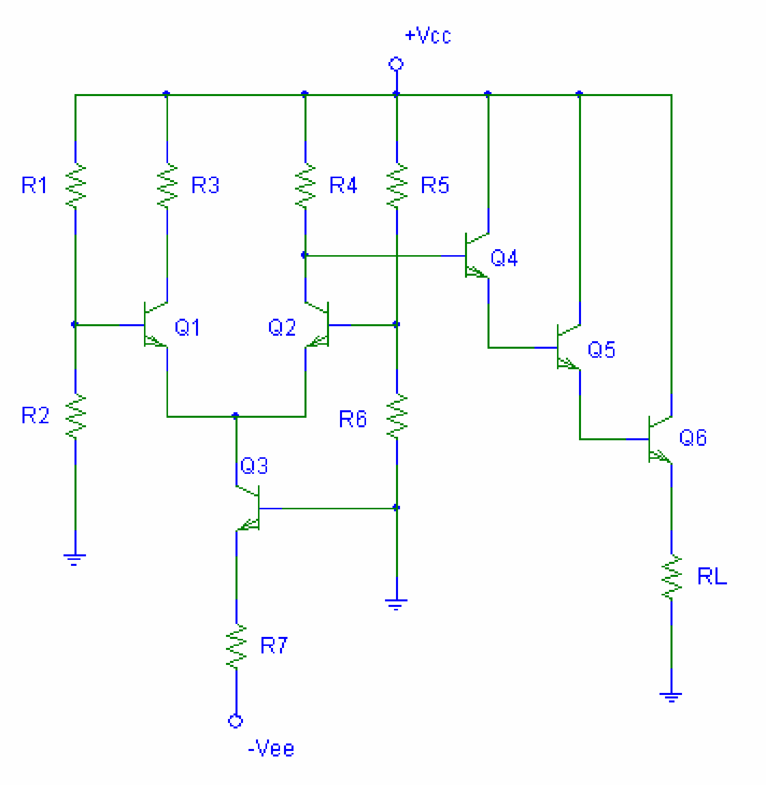


Figure 2.3: Circuit diagram for Exercise 2.2.

2.5. Could the above data be fitted by a rational function (equation 2.4 of section 2.4) having less number of coefficients than those of the polynomial approximation? [Try LS fit again.]

2.6. (a) Try to approximate the function f (x) = x^4 on the interval [0, 1] with only one polynomial piece. [A possible cubic spline function is $2x^3 – x^2$.]

(b) Again approximate f (x) = x^4 on [0, 1], but this time use two polynomial pieces so that x = [0, ½, 1], that is, use cubic splines with multiple (two) knots. Is there any improvement in the accuracy of representation?

2.7. Plot the function given in problem 2.4. Create an artificial neural network (ANN) with one input layer, one output layer and a hidden layer to approximate the above function. Choose proper weights for the links and biases for the neurons using a training set and back propagation algorithm (BEP). [Use MATLAB library routine for the purpose.]

2.8. Given below are 21 inputs P and associated target points T:
P = (–1, 0.1, +1)
T = (–0.9602, –0.5770, –0.0729, –0.3771, +0.6405, +0.6600, +0.4609, +0.1336, –0.2013, –0.4344, –0.5000, –0. 3930, –0.1647, +0.0988, +0.3072, +0.3960, +0.3449, +0.1816. –0.0312, –0.2189, −0.3201)
Define a suitable radial basis function (RBF) which could be used to fit the above target points. Now create a radial basis network which will approximate the function defined by P and T.

2.9. Try fitting in linguistic variables to describe the data given in problems 2.6 and 2.8 and define corresponding rule bases to explain the behaviour of the data.

References

[1] H. Walker and S.W. Director, “VLASIC: a catastrophic fault yield simulator for integrated circuits”, IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. CAD-5, pp. 541-556, October 1986.

[2] R.J.A. Harvey, A.M.D. Richardson, E.M.F.G. Bruls and K. Baker, “Analogue fault simulation based on layout dependent fault models”, Proceedings, 1994 IEEE International Test Conference, pp. 641-649, October 1994.

[3] T. Olbrich, J. Perez, I.A. Grout, A.M.D. Richardson and C. Ferrer, “Defect-oriented vs schematic-level based fault simulation for mixed-signal ICs, Proceedings, 1996 IEEE International Test Conference, pp. 511-520, October 1996.

[4] F. Fantini and C. Morandi, “Failure modes and mechanisms for VLSI ICs - a review”, IEE Proceedings - G, Circuits, Devices and Systems, Vol. 132, pp. 74-81, June 1985.

[5] F. Jensen, Electronic Component Reliability, John Wiley & Sons Ltd., West Sussex, 1995.

[6] A. Birolini, Quality and Reliability of Technical Systems, Springer-Verlag, Berlin, 1994.

[7] F.J. Langley, “ Printed Circuits Handbook”, Testing in Assembly, (Clyde F. Coombs, Jr., et al., eds.), McGraw-Hill Publishing Company, Inc., New York, pp. 21.1-21.26,

[8] E. Amerasekera and F. Najm, Failure Mechanisms in Semiconductor Devices, 2nd edition, John Wiley & Sons Ltd., West Sussex, 1997.

[9] W. Maly, A.W. Strojwas and S.W. Director, “VLSI yield prediction and estimation: A unified framework”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. CAD-5, pp. 114-130, January 1986.

[10] S.W. Director, W. Maly and A.W. Strojwas, VLSI Design for Manufacturing: Yield Enhancement, Kluwer Academic Publishers, The Netherlands, 1990.

[11] P. Duhamel and J.C. Rault, “Automatic test generation techniques for analog circuits and systems: A review”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 441-440, July 1979.

[12] M.J. Marlett and J.A. Abraham, "DC_ITAP - an iterative analog circuit test generation program for generating DC single pattern test", Proceedings, 1988 IEEE International Test Conference, pp. 839-844, October 1988.

[13] S.J. Tasi, "Test vector generation for linear analog devices", Proceedings, 1990 IEEE International Test Conference, pp. 592-597, October 1990.

[14] N. Nagi, A. Chatterjee, A. Balivada and J.A. Abraham, "Fault-based automatic test generator for linear analog circuits", Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-93), pp. 88-91, November 1993.

[15] G. Devarayanadurg and M. Soma, "Analytical fault modeling and static test generation for analog circuits", Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-94), pp. 44-47, November 1994.

[16] G. Devarayanadurg and M. Soma, "Dynamic test signal design for analog integrated circuits", Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-95), pp. 627-630, November 1995.

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

[18] S. Mir, M. Lubaszewski and B. Courtois, "Fault-based ATPG for linear analog circuits with minimal size multifrequency test sets", Journal of Electronic Testing: Theory and Application, Vol. 9, pp. 42-57, August/October 1996.

[19] A. Abderrahman, E. Cerny and B. Kaminska, "Optimization-based multifrequency test generation for analog circuits", Journal of Electronic Testing: Theory and Applications, Vol. 9, pp. 59-73, August/October 1996.

[20] L. Milor and V. Visvanathan, "Detection of catastrophic faults in analog integrated circuits", IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, Vol. 8, pp. 114-130, February 1989.

[21] B.R. Epstein, M. Czigler and S.R. Miller, "Fault detection and classification in linear integrated circuits: an application of discrimination analysis and hypothesis testing", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 12, pp. 103-113, January 1993.

[22] Z. Wang, G. Gielen and W. Sansen, "Testing of analog circuits based on power supply current monitoring and discrimination analysis", Proceedings, 3rd Asian Test Symposium (ATS'94), pp. 126-131, November 1994.

[23] J. Arguelles and S. Bracho, "Signature analysis for fault detection of mixed-signal ICs based on dynamic power-supply current", Journal of Electronic Testing: Theory and Application, Vol. 9, pp. 89-107, August/October 1996.

[24] C-Y. Pan, K-T. Cheng and S. Gupta, "A comprehensive fault macro-model for Opamps", Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-94), pp. 344-348, November 1994.

[25] C-Y. Pan and K-T. Cheng, "Pseudo-random testing and signature analysis for mixed-signal circuits", Digest of Papers, IEEE/ACM International Conference on Computer-Aided Design (ICCAD-95), pp. 102-107, November 1995.

[26] C-Y. Pan, K-T. Cheng and S. Gupta, "Fault macro-modeling and a test strategy for Opamps", Journal of Electronic Testing: Theory and Application, Vol. 9, pp. 225-235, December 1996.

[27] M. Can, "Modeling analog chips as if they are digital", Proceedings, 2000 IEEE Autotestcon (IEEE Systems Readiness Technology Conference), pp. 197-204, September 2000.

[28] D.S.S. Bello, R. Tangelder and H. Kerkhoff, “Modeling a verification test system for mixed-signal circuits”, IEEE Design & Test of Computers, Vol. 18, pp. 63-71, January-February 2001.

[29] G. Chiorboli and C. Morandi, “ADC modeling and testing”, Proceedings, 18th IEEE Instrumentation and Measurement Technology Conference (IMTC 2001), Vol. 3, pp. 1992-1999, May 2001.

[30] M. Sachdev, “A realistic defect oriented testability methodology for analog circuits”, Journal of Electronic Testing: Theory and Applications, Vol. 6, pp. 265-276, June 1995.

[31] N. Nagi and J.A. Abraham, “Hierarchical fault modeling for analog and mixed-signal circuits”, Digest of Papers, 1992 IEEE VLSI Test Symposium (10th Anniversary, Design, Test and Application: ASICs and System-on-a-Chip), pp. 96-101, April 1992.

[32] J. Galiay, Y. Crouzet and M. Vergniault, “Physical versus logical fault models MOS LSI circuits: Impact on their testability”, IEEE Transactions on Computers, Vol. C-29, pp. 527-531, June 1980.

[33] M.J.D. Powell, Approximation Theory and Methods, Cambridge University Press, Cambridge, 1981.

[34] L.L. Schumaker, Spline Functions: Basic Theory, John Wiley & Sons, Inc., New York, 1981.

[35] B. Kosko, Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence, Prentice-Hall, Inc., New Jersey, 1991.

[36] R.P. Lippmann, “An introduction to computing with neural networks”, IEEE ASSP Magazine, Vol. 4, pp. 4-22, April 1987.

[37] D.R. Hush and B.G. Horne, “Progress in supervised neural networks - what’s new since Lippmann”, IEEE Signal Processing Magazine, Vol. 10, pp. 8-39, January 1993.

[38] S-Y. Hsu, T. Masters, M. Olson, M. Tenorio and T. Grogan, “Comparative analysis of five neural network models”, Remote Sensing Reviews, Vol. 6, pp. 319-329, January 1992.

[39] M.S. Dawson and A.K. Fung, “Neural networks and their applications to parameter retrieval and classification”, IEEE Geoscience and Remote Sensing Society Newsletter, pp. 6-14, September 1993.

[40] S. Naidu, E. Zafifiou and T.J. McAvoy, “Use of neural networks for sensor failure detection in a control system”, IEEE Control Systems Magazine, Vol. 10, pp. 49-55, April 1990.

[41] T. Sorsa and H.N. Koivo, “Application of artificial neural networks in process fault diagnosis”, Automatica, Vol. 29, pp. 843-849, July 1993.

[42] B.J. Kagle, J.H. Murphy, L.J. Koos and J.R. Reeder, “Multi-fault diagnosis of electronic circuit boards using neural networks”, Proceedings, 3rd International Joint Conference on Neural Networks (IJCNN’90), Vol. 2, pp. 197-202, June 1990.

[43] W. Wiitanen, “Signature analysis: a general neural network application in process monitoring”, Presented at the SME Conference on Neural Network Applications for Manufacturing Product/Process Control, April 1991.

[44] R. Spina and S.J. Upadhyaya, “Linear circuit fault diagnosis using neuromorphic analyzers”, IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 44, pp. 188-196, March 1997.

[45] Alok Barua, Prithviraj Kabisatpathy and Satyabroto Sinha, “A method to diagnose faults in analog integrated circuits using artificial neural networks with pseudorandom noise as stimulus”, in Proceedings of 10th IEEE International Conference on Electronics, Circuits and Systems, Sharjah, U. A. E., December 14-17, 2003, pp. 356-359.

Chapter 3

TEST STIMULUS GENERATION

3.1 Introduction

With the growing importance of highly complex very large scale integrated (VLSI) circuits and systems-on-chips (SOCs), combined with the demand for shorter design and manufacturing cycles, the need for economical, fast and accurate test and fault diagnosis methods are apparent. At present, these ICs, especially the analog one, are tested using *ad hoc* or unstructured test methods on a wide assortment of expensive analog and digital test equipment. Moreover, consumers today demand high performance and quality in the electronic components that they buy. Low prices and years of problem-free operation with minimal maintenance are now the norm. In order for manufacturers to deliver such products, an extensive testing and fault diagnosis program must be practised [1][2]. This is to ensure that only good products are delivered to the consumer and those bad parts are either sent for repair or discarded. Therefore it is extremely important that all the ICs are tested before those go in the assembly line. Distributing the testing throughout each stage of manufacture (i.e., at wafer, die, board and system assembly) will minimise the cost incurred by testing [3]. A commonly mentioned rule of thumb of test is the *rule of ten* which suggests that the cost of detecting a bad component in a manufactured part increases tenfold at each level of assembly. Thus, discovering its presence early is most desirable. The question that emerges at this point is what kind of test should be performed; after all, testing does consume resources and takes time to perform. The answer lays with the observation that design errors are unrelated to those caused by manufacturing. Thus, the tests required to determine whether a design is acceptable can be very different from those required to determine whether the design has been manufactured correctly.

Testing has become an important issue in the production process of each electronic system, board or VLSI chip. Although testing incurs a lot of efforts it is finally an important means to reduce overall cost significantly. Another, equally important motivation for testing comes from the area of dependable computing. The increasing level of integration results in small feature size, small charges in the storage elements and high proximity of functional units. This not only requires extreme care in the chip layout and manufacturing process, but also makes the circuits highly susceptible to external faults. The trend towards high clock frequency and low power supply further aggravates the situation. At the same time computers are increasingly used for safety-critical applications (e.g. in power plants, transportation systems and medicine), where a system failure may have disastrous consequences. Testing has been recognised as a valuable means to (a) check system installation and configuration after maintenance activities, (b) ensure correct system functionality at start-up, and (c) avoid masking and accumulation of errors during operation. *Availability of a system (or of redundant system components) can be significantly increased, if testing is employed to allow rapid diagnosis after a failure*. These facts clearly show the economic potential of an efficient test strategy in the rapidly growing area of dependable systems.

One problem central to testing is the determination of an optimal test pattern under the following essential requirements:

- detection of (ideally) all defects assumed in the fault model,
- ease of generation/storage (low overhead)
- compactness (short test duration)

Combined with the above, the large number of specifications of analog circuits along with their complex nature enhances the requirement of elaborate signal sources and parameter extraction techniques. Extensive discussions on digital test pattern generation and the associated problems can be found in the literature [4][5][6][7][8][9][10]. Below is given a brief overview of the principles.

3.1.1 Type of test signal: deterministic versus pseudo-random

The determination of a test pattern usually involves feedback from a fault model. In a first step a list of all faults considered by the fault model is made. In the second step a test pattern is assumed and all faults detected by it are removed from the fault list. Repeating this step for new test patterns progressively reduces the fault list. Towards the end of the iteration process the contribution of new pattern decreases, since the fault list becomes small. One new pattern may be needed to remove one single fault from the list, while other new patterns do not make any contribution at all. Although an exhaustive application of this *deterministic algorithm* (D-algorithm [11]) promises the detection of all detectable faults, the duration of the search process and the length of the resulting test pattern may be excessive. A balance between coverage and cost must be found at this point.

A special case of the deterministic test is the *exhaustive test*, for which *all* possible test patterns are applied. While this test produces the best possible coverage, it is impracticable for a complete VLSI chip. Considering the partitioning method a sum of exhaustive tests can be applied progressively to all parts of the circuit. This method is called *pseudo-exhaustive test* [5][12]. Test pattern generation for the exhaustive and pseudo-exhaustive test is trivial and does not involve fault simulation.

In a completely different approach a fixed number of test patterns are generated without feedback from the fault model. The sequence of test patterns is called *pseudo-random*, because it has some important properties of a random sequence, while being totally predictable and repeatable [12]. The coverage of the test sequence is checked by fault simulation. This process of analysing coverage via a fault simulation is called *fault grading* [13]. If a sufficient coverage level has been reached, the set is accepted; otherwise a new one is generated. One drawback of this *pseudo-random algorithm* is that the required length of the test sequence is often hard to determine a priori. Statistical frameworks have been derived for this purpose [12]. Although a sequence found by this approach does not necessarily have the best test coverage possible, it has advantages over the deterministic search:

- For a reasonable coverage limit the determination of a sequence is extremely fast.
- The random patterns can easily be generated on-chip by a hardware random generator.
- Random patterns have been shown to have much better coverage for non-target defects, i.e., faults that are not considered by the fault model [14][15].
- Since cost is an extremely critical issue in every design, the deterministic approach may have to be terminated at a quite early iteration stage. In this case the results of the random approach with comparable cost may be superior.

The methods for test pattern generation are highly correlated with the types of test pattern discussed above. In the following sections some common approaches are discussed which may be suitable for analog test stimulus generation.

3.2 Conventional analog test stimulus generation

In general, analog testing and fault diagnosis problem may be broken down in two aspects. First, the circuit under test (CUT) must be simulated using a suitable test stimulus generator and then the output of the CUT must be analysed. This and the following sections will address the problems of test stimulus generation in analog circuits.

There are two distinctly different approaches for the generation of the most commonly used test stimulus like the sinusoidal waveform, and the square, triangular, pulse, etc. waveforms. The first approach employs a positive feedback loop consisting of an amplifier and an RC or LC frequency selective network. These circuits which generate sinusoidal waveforms utilising resonance phenomena are known as linear oscillators. Circuits that generate the square, triangular, pulse, etc. waveforms are called nonlinear oscillators or function-generators. Most commonly used sinusoidal oscillators discussed in this chapter employ operational amplifier (Op-Amp) and is intended for precision analog applications.

3.2.1 Sinusoidal oscillator circuits

The basic structure of a sinusoidal oscillator consists of an amplifier and a frequency selective network connected in a positive feedback loop [16]. A generalised block diagram of such a circuit is as shown in Figure 3.1.

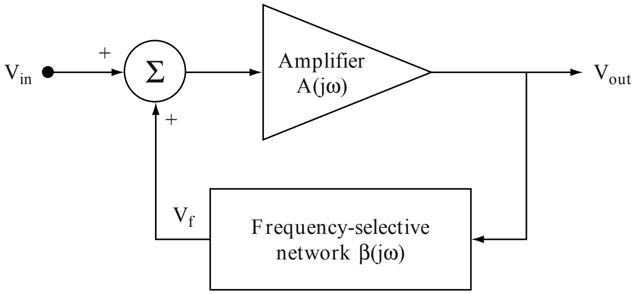


Figure 3.1: Block diagram of a linear feedback oscillator.

Although in an actual oscillator circuit no input signal is present, Figure 3.1 shows an input signal v_{in} to help explain the principle of operation. According to the *Barkhausen criterion*, the system will sustain oscillation irrespective of the input v_{in} if, at a particular frequency ω_0, the loop gain $A(j\omega)\beta(j\omega)$, which corresponds directly to the actual gain seen around the feedback loop, is equal to unity and the phase of the loop gain should be zero. Mathematically, this can be written as

$$A(j\omega_0)\beta(j\omega_0) = 1.0 \tag{3.1}$$

and $phase_{loop} = \phi_A + \phi_\beta = 0$ (3.2)

It is important to note that the circuit of Figure 3.1 to oscillate at one frequency the *Barkhausen criterion* should be satisfied at one frequency only (that is, ω_0); otherwise the resulting output waveform will not be a simple sinusoid.

3.2.1.1 The Wien bridge oscillator

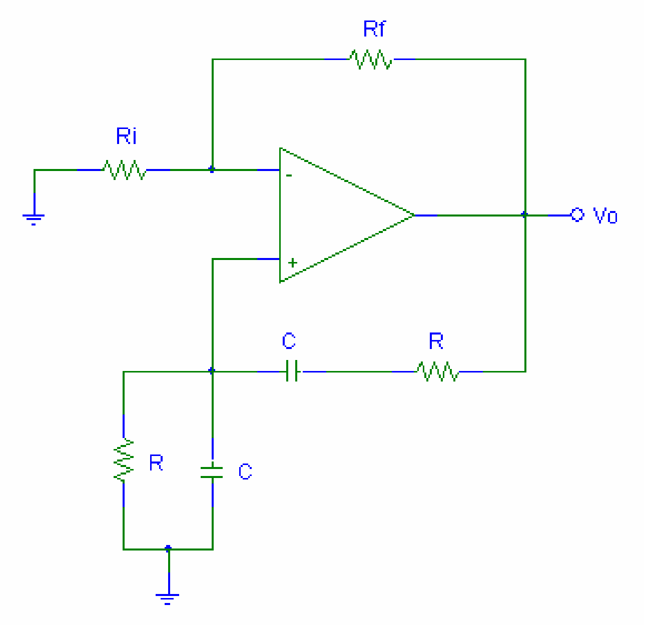


Figure 3.2: The Wien bridge oscillator.

One of the simplest oscillator circuit based on the above principle is the Wien bridge oscillator shown in Figure 3.2. The circuit consists of an Op-Amp connected in the non-inverting mode with a closed loop gain of $1 + R_f / R_i$. In the feedback path an RC network is connected.

The loop gain $A(j\omega)\beta(j\omega)$ of the circuit can be easily calculated by multiplying the transfer function of the feedback RC network with the amplifier gain as

$$A(j\omega)\beta(j\omega) = \frac{1 + R_f / R_i}{3 + j(\omega CR - 1/\omega CR)} \tag{3.3}$$

The loop gain will be a real number (that is, the phase will be zero) at one frequency only and is given by

$$\omega_0 = 1/CR \tag{3.4}$$

To obtain sustained oscillations at this frequency, the magnitude of the loop gain should be set to unity. It can be achieved by selecting $R_f / R_i = 2$. To ensure that oscillation will start, the ratio of R_f / R_i is chosen slightly greater than 2. It can be easily verified that if $R_f / R_i = 2 + \delta$, where δ is a small positive number, the roots of the characteristic equation $(1 - A(j\omega)\beta(j\omega))$ of the circuit of Figure 3.2 will always be in the right-half of the s-plane. However the circuit is to be stabilized using nonlinear element along with frequency selective components.

3.2.1.2 The RC phase shift oscillator

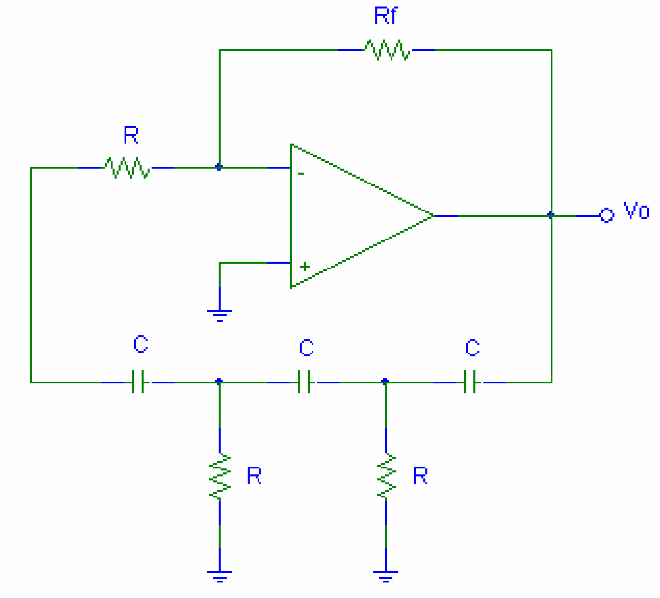


Figure 3.3: The RC phase shift oscillator.

A basic RC phase shift oscillator is shown in Figure 3.3. It consists of a negative gain amplifier with a three section RC ladder network in the feedback loop. Each section will produce a phase shift of 60^0. This circuit will oscillate at the frequency for which the total phase shift of the RC network is 180^0. It is important to note that only at this frequency the total phase shift around the loop will be 0 or 360^0 provided there will be no loading effect of the network on the amplifier.

For sustained oscillation, the value of the gain of the Op-Amp should be equal to the inverse of the magnitude of the RC network transfer function at the frequency of oscillation. However, in order to ensure that oscillation starts, the gain of the Op-Amp is chosen slightly higher than the value that satisfies the unity-loop-gain condition. Here closed loop gain of the Op-Amp will be equal to R_f/R that must be greater than 29. Then the oscillation will grow and ultimately it is be limited by some nonlinear elements. The frequency of oscillation becomes

$$f_0 = \frac{1}{2\pi RC\sqrt{6}} \tag{3.5}$$

3.2.1.3 The Tuned oscillator circuits

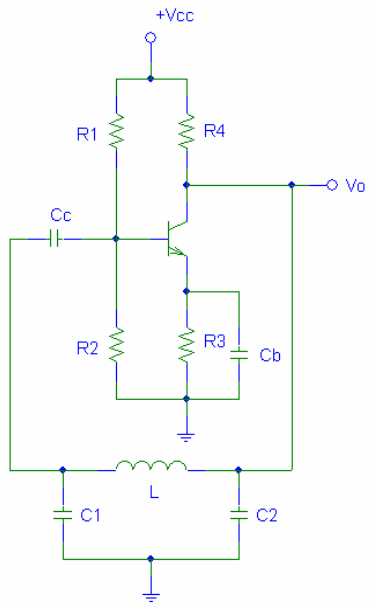


Figure 3.4: The Colpitts oscillator.

Tuned oscillator circuits generate stable sinusoidal waveforms. One of the popular configurations, known as the Colpitts oscillator is shown in Figure 3.4. The LC tuned circuit has higher Q than RC types however the oscillators are difficult to tune over wide range of frequencies. The frequency of oscillation, ω_0, of this circuit can be determined by the resonant frequency of the LC tank circuit and is given by

$$\omega_0 = \sqrt{\frac{1}{L}\left(\frac{1}{C_1}+\frac{1}{C_2}\right)} \tag{3.6}$$

The ratio C_1/C_2 determines the feedback factor and it must be adjusted in conjunction with the gain of the amplifier to ensure that oscillation starts. Alternatively, a piezo-electric crystal whose electrical equivalent circuit is shown in Figure 3.5 may be used in place of the inductor. The resulting circuit has an oscillation frequency very close to the parallel resonance frequency ω_p of the crystal. This frequency can be expressed as (3.7)

$$\omega_p = \sqrt{\frac{1}{L_0}\left(\frac{1}{C_0}+\frac{1}{C_1}\right)}$$

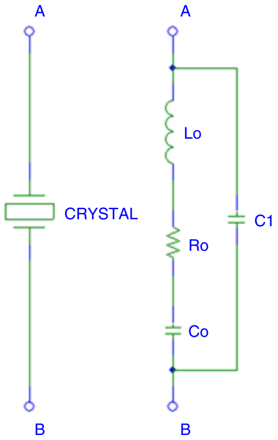


Figure 3.5: ***The electrical equivalent circuit of the piezoelectric crystal.***

All the circuits discussed above are capable of generating stable sinusoidal signals over a broad band of frequency. But they require bulky circuit components which are not compatible to monolithic IC realisation. Besides, the frequency of oscillation ω_0 is not easily adjustable.

3.3 Digital test stimulus generation

Developments in analog-to-digital (A/D) and digital-to-analog (D/A) converter technology has motivated the IC designers to implement analog signal generation using digital signal processing (DSP) techniques. The same philosophy can be extended to analog test stimulus generation. Use of digital frequency synthesis and lossless discrete integrator approach are examples of analog signal generation using digital techniques.

3.3.1 Digital frequency synthesis

Using a cascade of a digital signal source and a digital-to-analog (D/A) converter analog sinusoidal signals can be generated [17][18][19]. Such circuits are known as digital frequency synthesisers (DFS). A basic DFS architecture is shown in Figure 3.6.

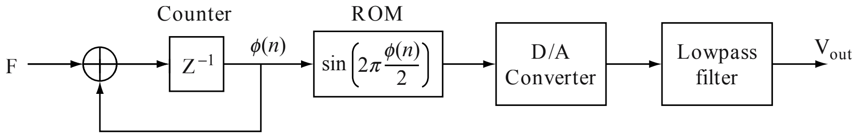


Figure 3.6: Basic architecture of a digital frequency synthesiser.

In the DFS shown in the figure above an L-bit counter is used as an integrator that increments the phase variable $\phi(n)$. A frequency control constant, F, is used to generate the phase argument, $\theta(n)$, of a sinusoidal computation. The relationship between $\theta(n)$ and $\phi(n)$ is given by

$$\theta(n) = 2\pi\left(\frac{\phi(n)}{2^L}\right) \tag{3.8}$$

The periodicity of the waveform is accounted for by the overflowing property of the modulo-2^L counter. The output is a digital sine wave which is converted to an analog signal by passing it through the D/A converter and a low pass filter. The output frequency of the DFS is given by [20]

$$f_{out} = \frac{\phi(n) - \phi(n-1)}{2\pi\Delta t_{clk}} = \frac{f_{clk}F}{2^L} \tag{3.9}$$

The frequency resolution of the DFS is a function of the number of samples stored in the read only memory (ROM) while the signal quality is limited either by word length of each sample or the characteristics of the D/A converter. The drawback of the ROM based DFS is that the silicon area required to implement the circuit is usually very large.

3.3.2 Lossless discrete integrator

Many analog and digital filter design techniques are based on the LC ladder filter structure because of their exceptional sensitivity properties. One approach, the lossless discrete integrator (LDI) [21], based on the passive LC ladder filter has been successfully applied to switched capacitor and digital filter circuits [22][23].

Turner and Ramesh have provided a complete synthesis procedure for the design of digital filters based on LC ladder networks [23]. The structure of the resulting digital filter is generally formed from a set of coupled second order resonators. These resonators are formed by cascading two discrete-time integrators of the form $[z^{-1}/(1-z^{-1})]$ and $1/(1-z^{-1})$ in a loop with the sign of one integrator being positive and the other negative. Such an arrangement is shown in Figure 3.7.

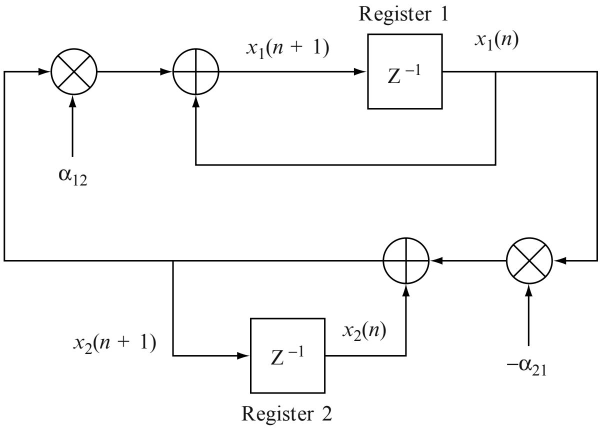


Figure 3.7: A second-order digital resonator consisting of a cascade of two integrators in a loop.

A digital oscillator may be realised through elimination of the damping in the filter as a special case of the resonant circuit shown in Figure 3.7. To analyse this circuit let us consider the analog counterpart, the LC tank circuit, shown in Figure 3.8. Being a lossless circuit, it follows that once the LC tank circuit is excited, no energy is lost but instead alternates between electric and magnetic forms. The resulting capacitor voltage and inductor current waveforms represent ideal sinusoids with frequency $\omega_0 = 1/\sqrt{LC}$.

It is worthwhile to mention here the two important characteristics of the energised LC tank circuit that gives an insight to the behaviour of the second order digital resonator. These are:

1. variation in the capacitor or inductor value does not prevent the circuit from oscillating, but merely shifts the frequency of oscillation.
2. the amplitude of oscillation is a function of the initial conditions imposed on the capacitor and inductor.

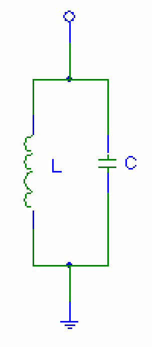


Figure 3.8: A parallel LC tank circuit.

Now comparing the above mentioned characteristics of LC tank circuit with the behaviour of the second order digital resonator of Figure 3.7 it can be concluded that

1. the variations in the coefficients α_{21} and α_{12} (corresponding to L and C of the tank circuit) may cause shifts in the oscillating frequency but will not prevent the circuit from oscillating, and
2. the amplitude of the oscillatory tone will be a function of the initial conditions imposed on the two discrete-time integrators.

These can be easily verified from the study of the characteristic equation of the resonator. Denoting the values at the output of the two discrete-time integrators at time $t = nT$ by $x_1(n)$ and $x_2(n)$ respectively, the two difference equations characterising the digital resonator can be written as,

$$x_1(n+1) = x_1(n) + \alpha_{12} x_2(n+1) \tag{3.10}$$

$$x_2(n+1) = -\alpha_{21}x_1(n) + x_2(n) \tag{3.11}$$

From the above two equations, $x_2(n)$ can be eliminated to give a single equation in $x_1(n)$. Taking the z-transform of the resultant equation, with $X_1(z)$ representing the z-transform of $x_1(n)$, one obtains

$$z^2 X_1(z) + (\alpha_{12}\alpha_{21} - 2)zX_1(z) + X_1(z) = 0 \tag{3.12}$$

The characteristic equation is then given by

$$z^2 + (\alpha_{12}\alpha_{21} - 2)z + 1 = 0 \tag{3.13}$$

The location of the poles may be determined by finding the roots of the above equation, which are

$$z_{1,2} = (1 - \frac{\alpha_{12}\alpha_{21}}{2}) \pm \frac{1}{2}\sqrt{\alpha_{12}\alpha_{21}(\alpha_{12}\alpha_{21} - 4)} \tag{3.14}$$

The discriminant of the above expression will always be negative, yielding complex roots, if the product $\alpha_{12}\alpha_{21}$ is restricted to values between 0 and 4. Moreover, the equation for the two roots $z_{1,2}$ can be written in the polar notation, respectively for 0<$\alpha_{12}\alpha_{21}$<2, corresponding to roots in the right-half plane and for 2<$\alpha_{12}\alpha_{21}$<4, corresponding to roots in the left-half plane as,

$$z_{1,2} = e^{\pm j\cos^{-1}(1-\frac{\alpha_{12}\alpha_{21}}{2})} \tag{3.15}$$

$$z_{1,2} = e^{\pm j(\pi-\cos^{-1}(1-\frac{\alpha_{12}\alpha_{21}}{2}))} \tag{3.16}$$

It is evident from the above two equations that for guaranteed oscillation, the roots of the characteristic equation must lie on the unit circle for all values of the product $\alpha_{12}\alpha_{21}$ between 0 and 4. In addition, the frequency of oscillation, ω_0, of the digital resonator can be obtained directly from the phase terms of the above two equations. If the digital resonator is clocked at a rate of $f_{ck} = \frac{1}{T}$, the equation for ω0 will be given as,

$$\omega_0 = \begin{cases} f_{ck} \cos^{-1}(1 - \frac{\alpha_{12}\alpha_{21}}{2}) & \text{for } 0 < \alpha_{12}\alpha_{21} < 2 \\ f_{ck}[\pi - \cos^{-1}(1 - \frac{\alpha_{12}\alpha_{21}}{2})] & \text{for } 2 < \alpha_{12}\alpha_{21} < 4 \end{cases} \quad (3.17)$$

It can now be concluded from the above equation that the oscillation frequency of the digital resonator varies continuously between 0 and $f_{ck}/2$ for values of $\alpha_{12}\alpha_{21}$ between 0 and 4. Realisation of a digital oscillator using the lossless discrete integrator explained above has been reported by Turner [24].

3.4 Delta sigma (Δ-Σ) signal generation

Delta sigma oscillators generate spectrally pure analogue signals. Due to their small silicon area requirement and digital implementation, delta-sigma oscillator has been identified as an important building block for BIST schemes. This class of circuit is based on digital resonators where frequency control is implemented with a multiplication operation. However a $(N\ bit)\, by\, (N\ bit)$ digital multiplier is slow when compared to adders and it occupies a large area on a die. A fixed-length multiplication is illustrated in Figure 3.9 with a co-efficient ***c*** and a time-varying signal B.

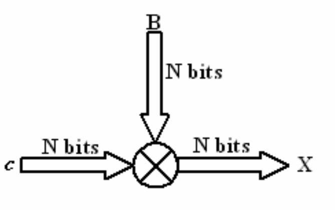


Figure 3.9: Fixed length multiplication.

If the scalar can be made a power-of-two or a sum of a few powers-of-two, then the use of multipliers can be avoided. In many applications however, this restriction is unbearable as the location of circuit poles and zeros can be very sensitive to such quantization.

3.4.1 Delta sigma (Δ-Σ) attenuator

The 1 bit encoding property of Δ-Σ modulators [25] can be used to simplify the multiplier, reducing it to a $(N\ bit)\, by\ (1\ bit)$ multiplication. This operation can then be implemented with a 2 to 1 multiplexer as shown in Figure 3.10. A necessary condition for this combination to work as a multiplier, known as a Δ-Σ attenuator [26], is that the Δ-Σ modulator must have a unity signal transfer function. In other words, the gain must be unity and have 0° phase shift, at least in the signal band. While this is not a property of most Δ-Σ modulators, there are topologies that will ensure this without constraining the designer in the choice of bandwidth or signal-to-noise ratio (SNR) [27].

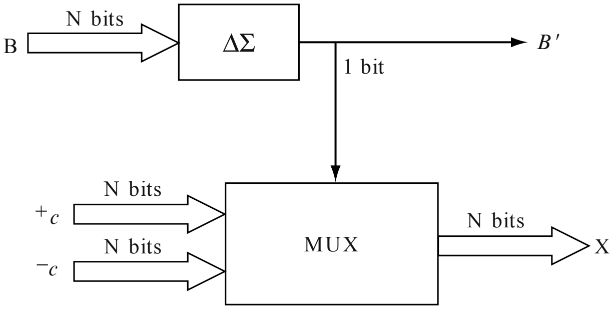


Figure 3.10: Delta sigma attenuator.

The time-varying signal (B) is now available in 1 bit form *(B')* at the output of the Δ-Σ modulator but the original signal is superimposed by an out-of-band quantization noise which can be eliminated later by filtering. Since the 1 bit DACs are compact and show excellent linearity, this can be used to convert the 1 bit representation of a digital signal easily to the analog domain. Because the quantization noise is introduced into the system, one must take sufficient care to use a Δ-Σ attenuator in the feedback loop. This can be achieved by making the coefficient ***c*** small and much less than unity in order to maintain the noise power to low levels.

3.4.2 Low pass delta sigma (Δ-Σ) oscillator

Lu et al [28] introduced a high quality analog oscillator using Δ-Σ modulator in the feedback loop of a digital resonator, known as the low pass Δ-Σ oscillator. The circuit, shown in Figure 3.11, is composed of two digital integrators, a low pass Δ-Σ modulator and a 2 to 1 multiplexer. The output of the Δ-Σ modulator is passed through a low pass filter to obtain a precision analog signal. The Δ-Σ modulator encodes its input on a 1 bit stream and the multiplexer realises a $(1\ bit)\ by\ (N\ bit)$ multiplication. To achieve the operation of a digital multiplier, the low pass Δ-Σ modulator must have a unity signal transfer function (STF). The circuit may also accommodate a low pass Δ-Σ modulator that has an STF equal to z^{-1} (such as the classical second-order low pass Δ-Σ modulator [29] shown in Figure 3.12) but the resonator must be modified by changing the delayed integrator for a non-delayed type.

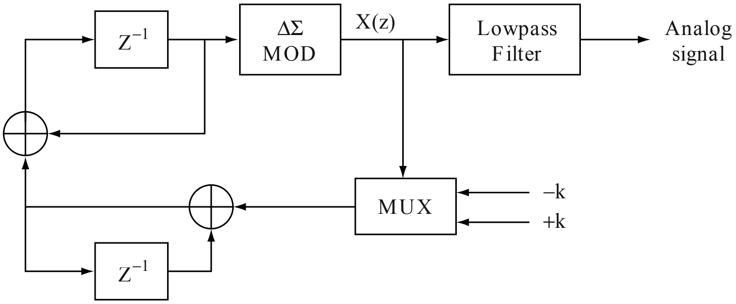


Figure 3.11: Low pass delta sigma oscillator.

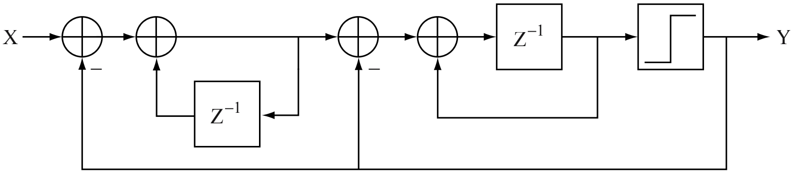


Figure 3.12: Second order low pass delta sigma modulator.

The characteristic equation of the circuit shown in Figure 3.11 is found to be

$$z^{-2} - (2-k)z^{-1} + 1 = 0 \qquad (3.18)$$

For $0 \le k \le 4$ in the above equation, the poles will lie on the unit circle of the z-plane resulting in oscillation. The frequency of oscillation (f_o) can be related to k and the clock frequency (f_{ck}) through the relation

$$\Omega_0 = \frac{2\pi f_0}{f_{ck}} = \cos^{-1}\left(1 - \frac{k}{2}\right) \qquad (3.19)$$

where Ω_0 is the normalised angular frequency expressed in radians. Frequency analysis of the 1–bit signal at the circuit output reveals that it contains a very high-quality low-frequency analog sine wave corrupted with high frequency noise. The analog sine wave signal can now be separated through low pass filtering. The amplitude and phase of the sine wave depend on the initial contents of the two registers, denoted as ϕ_1 and ϕ_2, respectively. Considering the ΔΣ modulator and the multiplexer combination as a perfect multiplier and using z-domain analysis, the following equation for the signal at the output of the ΔΣ modulator is obtained:

$$X(z) = \frac{(1 - z^{-1})\phi_1 + z^{-1}\phi_2}{1 - (2 - k)z^{-1} + z^{-2}} \tag{3.20}$$

Using the inverse z-transform, the time-domain signal description is obtained as:

$$x(nT) = \phi_1 \frac{\sqrt{2(1 - \cos\Omega_0)}}{\sin\Omega_0} \sin\left(\Omega_0 nT + \frac{\pi + \Omega_0}{2}\right) + \phi_2 \frac{1}{\sin\Omega_0} \sin(\Omega_0 nT) \tag{3.21}$$

where T is the clock period and n is a time index. The first term is the oscillation due to the initial condition ϕ_1 while the second term describes the sine wave triggered by the initial value of ϕ_2. The amplitude and phase of the sinusoidal signal can be selected separately since the two terms are linearly independent.

3.4.3 Band pass delta sigma (Δ-Σ) oscillator

The basic Δ-Σ oscillator was modified for band pass (BP) Δ-Σ modulation [30] resulting in the circuit shown in Figure 3.13. As can be seen, the circuit consists of a second-order resonator and a feedback loop comprising a band pass Δ-Σ modulator in series with a 2 to 1 multiplexer. The centre of the oscillation frequency range is set by the coefficient K_c and the fine tuning of the oscillation around this centre frequency is done by the coefficient k_f.

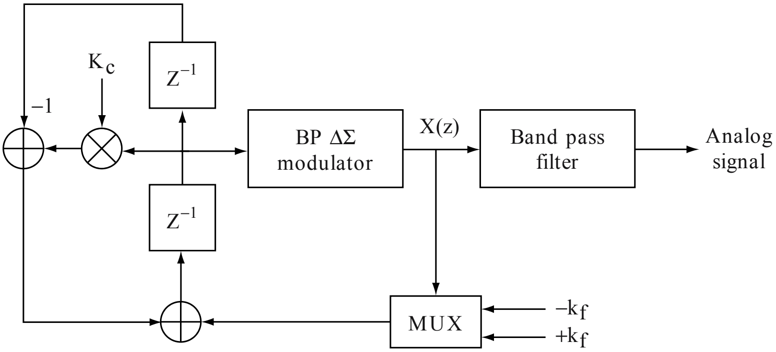


Figure 3.13: Band pass delta sigma oscillator.

It is reported in the literature [31] that for a smaller silicon area, K_c should be a power-of-two or a sum of a small number of power-of-two terms to avoid the need for a full-fledged multiplier. In particular, if the value of K_c = 0 is used the signal band is found to be located at a quarter of the clock frequency and results in the implementation of an economical BP Δ-Σ modulator. The 1 bit signal at the output of the Δ-Σ modulator contains the sine wave oscillatory signal together with the out-of-band quantization noise. The sine wave is separated by passing the bit stream through a band pass filter. It should be noted again that the Δ-Σ modulator must have a unity signal transfer function (STF).

The characteristic equation of the band pass Δ-Σ oscillator is given by

$$z^{-2} + (k_f - K_c)z^{-1} + 1 = 0 \qquad (3.22)$$

from which the equation for the frequency of oscillation can be obtained as:

$$\Omega_0 = \frac{2\pi f_0}{f_{ck}} = \cos^{-1}\left(\frac{K_c - k_f}{2}\right) \qquad (3.23)$$

The amplitude and phase of the sine wave depend on the initial contents of the two registers, denoted as ϕ_1 and ϕ_2, respectively. Using simple z-domain analysis, one can write the following equation for the BP ΔΣ modulator output signal:

$$X(z) = \frac{\phi_1 - \phi_2 z^{-1}}{1 + (K_c - k_f)z^{-1} + z^{-2}} \qquad (3.24)$$

from which the corresponding time-domain equation is found to be:

$$x(nT) = \phi_1 \frac{1}{\sin\Omega_0} \sin(\Omega_0 nT + \Omega_0) + \phi_2 \frac{1}{\sin\Omega_0} \sin(\Omega_0 nT) \quad (3.25)$$

The amplitude and phase of the signal can be selected independently as the two sinusoids have a Ω_0 phase difference. If the exact value of phase is not important, one can set $\phi_2 = 0$ and select the desired amplitude through ϕ_1. The stability of the circuit as well as the signal quality can be verified experimentally.

3.4.4 Multi-tone delta sigma (Δ-Σ) oscillator

In the measurement of analog circuit parameters such as inter-modulation distortion, etc. multi-tone signals are used. Lu and Roberts [32] have demonstrated a hardware efficient technique to realise a multi-tone signal generator by modifying the basic design of the Δ-Σ oscillator. They used time division multiplexing to interleave the bit streams resulting in a single bit stream at the output. This resulted in the reduction of the effective clock frequency by a factor equal to the number of tones. The circuit shown in Figure 3.11 was modified for two-tone signal generation using time-division multiplexing which is shown in Figure 3.14.

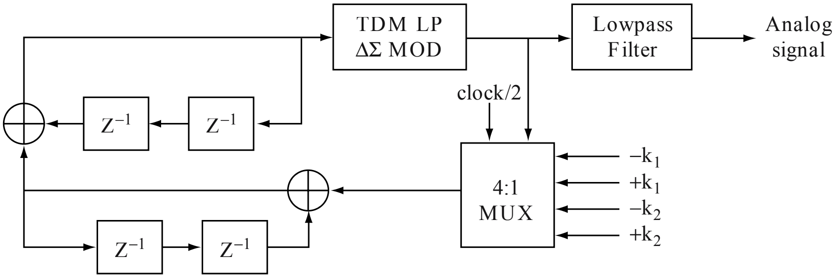


Figure 3.14: Two-tone low pass delta sigma oscillator.

Here the combinational part of the circuit is kept the same except that the 2:1 multiplexer is replaced by a 4:1 multiplexer and all registers are replaced by a string of two registers. However, as can be seen the content of the first of these two registers only has an effect on the combinational circuit on any clock cycle. In the following clock cycle, the content of the second one is just passed on to the first for its use.

Figure 3.15 shows how the second-order LP Δ-Σ modulator of Figure 3.12 can be modified to handle a two-tone time-division multiplexed signal for use in the multi-tone Δ-Σ oscillator.

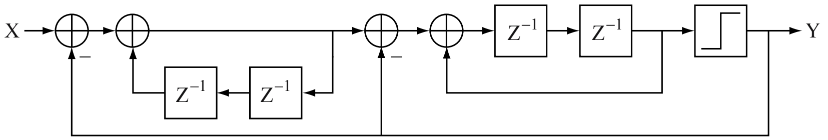


Figure 3.15: Time division multiplexed second order low pass delta sigma modulator.

Similar to the LP Δ-Σ oscillator, the BP Δ-Σ oscillator may be modified for multi-tone signal generation using time-division multiplexing. By applying this technique to the circuit of Figure 3.13 one obtains the BP Δ-Σ oscillator configuration shown in Figure 3.16.

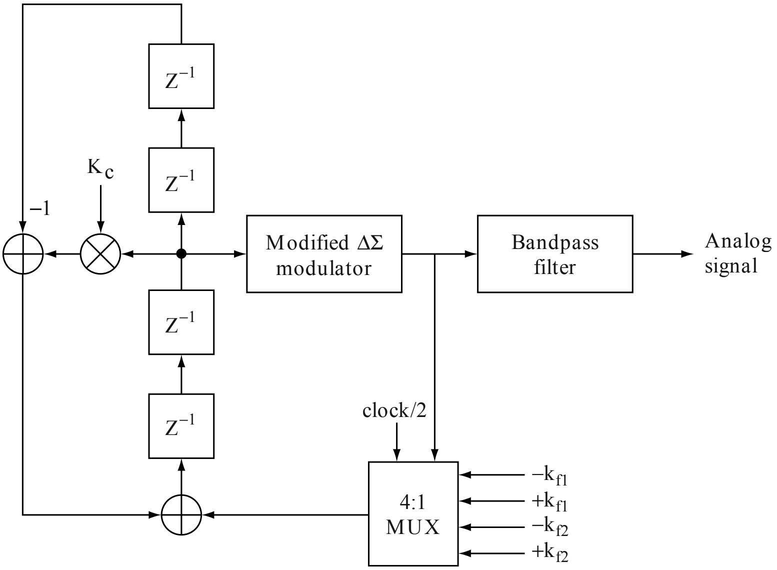


Figure 3.16: Two-tone band pass delta sigma oscillator.

In this section, a technique for generating high quality analog sinusoidal waveforms has been discussed. Except low pass reconstruction filter, the entire circuitry can be implemented digitally. As a result, the test stimulus generator is virtually unaffected by process variations and provides complete programmability over the sinusoidal amplitude, frequency and phase. However, an alternative to the Δ-Σ oscillators discussed above may be the pseudorandom noise generator discussed in the following section.

3.5 Pseudorandom noise generation

A piece-wise constant waveform with randomly distributed amplitudes can be generated by using a pseudorandom binary sequence (PRBS). Two basic parameters can be varied in the pulses of the PRBS: amplitude and width. Schreiber [33] used the expected response of the circuit to optimise the input test sequence by applying a sequence of pulses with varying amplitudes. Varying the amplitude of the pulses within a test sequence virtually rules out the use of digital testers. But it is easy to vary the width of a pulse with a digital tester. The PRBS is chosen to generate the pseudorandom noise (PRN) because it can be readily generated by a digital tester, have well defined properties and can be used to extract the impulse response of the IC as shown in Figure 3.17.

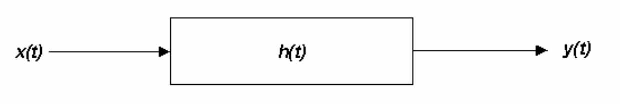


Figure 3.17: Block diagram of analog circuit testing.

The analog circuit being tested is represented by the block in Figure 3.17. $h(t)$, $x(t)$ and $y(t)$ are the impulse response, the input and the output of the circuit under test respectively. The output signal $y(t)$ is the convolution of the input $x(t)$ and the impulse response $h(t)$, given by,

$$y(t) = \int_{-\infty}^{\infty} h(\nu)x(t-\nu)d\nu \tag{3.26}$$

Cross-correlating $y(t)$ with $x(t)$ gives,

$$\phi_{xy}(\tau) = \int_{-\infty}^{\infty} x(t)y(t+\tau)dt \tag{3.27}$$

Substituting for $y(t)$ in Equation (3.27) from Equation (3.26), we get,

$$\begin{aligned}\phi_{xy}(\tau) &= \int_{-\infty}^{\infty} x(t)\int_{-\infty}^{\infty} h(\nu)x(t+\tau-\nu)d\nu dt \\ &= \int_{-\infty}^{\infty} h(\nu)\phi_{xx}(\tau-\nu)d\nu \end{aligned} \tag{3.28}$$

where, $\phi_{xx}(\tau - \nu)$ is the autocorrelation function of $x(t)$ and τ and ν are time shift constants.

Equation (3.28) states that the correlation between $x(t)$ and $y(t)$ is the convolution of the impulse response $h(\nu)$ with the autocorrelation of $x(t)$. If $x(t)$ is a white noise signal, then its autocorrelation function is a delta function and substituting this in Equation (3.28) we get,

$$\phi_{xy}(\tau) = \int_{-\infty}^{\infty} h(\nu)\delta(\tau - \nu)d\nu = h(\tau) \qquad (3.29)$$

as $\delta(\tau - \nu) = 1$ at $\tau = \nu$ and zero elsewhere.

Generation of white noise required in Equation (3.29) is not easy as white noise contains all frequencies with equal intensity, an infinite bandwidth and unlimited spectral energy. A piece-wise constant waveform with randomly distributed amplitudes can be generated by using a PRBS. The PRN signals generated from the PRBS signals have very good randomness properties and are very good approximations to white noise, in the frequency band of interest.

The simplest method for generating a PRBS is based on the linear feedback shift register (LSFR) [34]. The idea is to start with a register filled with some arbitrary pattern (not all zeros), then shift it right (say) a step at a time, filling in the vacated position from the left with a bit determined by the contents of the register. Figure 3.18 shows a simple five-bit LFSR, where a feedback path consisting of a modulo-2 sum of the third and fifth stage outputs is applied to the input of the first stage. The output sequence is obtained by examining the digits stored in the register's fifth stage as one progresses step by step through the various register stages. For this combination, the period p is $(2^5 - 1)$ = 31 and a 31-digit output sequence is the longest sequence one can generate with a five digit LFSR. The term linear refers to the modulo-2 sum operation in the feedback path. Sequences generated by this method are called the maximal length linear shift register (MLLSR) sequences, or m-sequence, in short, of which the PRN sequences are a special case. In general, an n-stage shift register with appropriate feedback connections is capable of generating an output sequence whose period p is $(2^n - 1)$ binary digits.

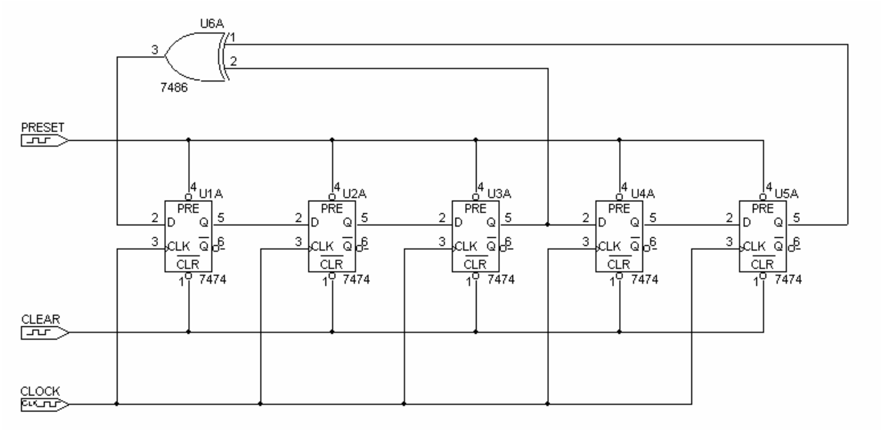


Figure 3.18: A five-stage maximum length linear feedback shift register.

Increasing the number of stages, i.e., *n* will have an interesting effect on the randomness of the generated sequence. This would be evident from a consideration of the auto-correlation function of the MLLSR sequence. Since the number of ones exceeds the number of zeros by one in each sequence, the auto-correlation function (ACF) becomes

$$R_{xx}(\tau) = \frac{1}{2^n - 1} \sum_{t_0}^{2^n - 1} f(t) f(t + nt_0) = -\frac{1}{p} \quad \text{for } t_0 \neq 0, T, 2T, \ldots \; etc$$

$$= +1 \text{ for } t_0 = 0, T, 2T, \ldots \; etc \quad (3.30)$$

where t_0 is the clock period, T is the repetition period of the m-sequence and the sequence is made bi-polar. The ACF has the form as shown in Figure 3.19. This ACF is the same as that of a real random binary sequence except for two differences: this sequence is not purely random, but periodic and the auto-correlation between peaks is not zero but has a small negative value.

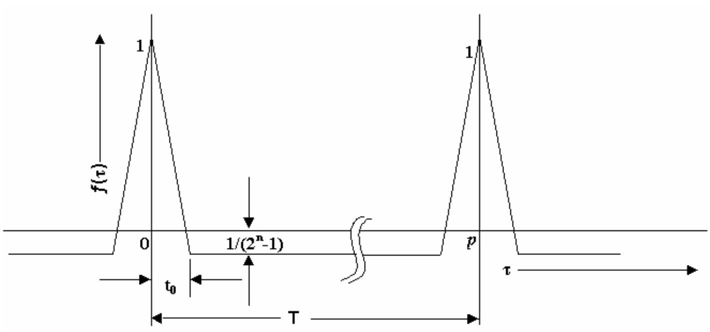


Figure 3.19: Auto-correlation function of maximum length linear shift register sequence.

Making *n* sufficiently large one can make both these differences as small as possible. The choice of n depends on the particular application of the noise generator. Generally, it may be stated that the period of the code has to be large enough compared to the response time of the circuit under test.

The power spectral density (PSD) of the m-sequence whose ACF is shown in Figure 3.19, is given by

$$S(\omega) = \frac{p+1}{p^2}\left[\frac{\sin\left(\frac{\omega t_0}{2}\right)}{\left(\frac{\omega t_0}{2}\right)}\right]^2 \sum_{\substack{n=-\infty \\ n\neq 0}}^{n=\infty} \delta\left(\omega - \frac{2\pi n}{p t_0}\right) + \frac{1}{p^2}\delta(\omega) \qquad (3.31)$$

where p is the period of the sequence, and t_0, the clock period of the waveform. The PSD for a typical m-sequence is shown in Figure 3.20.

It is still a line spectra with frequencies at multiples of the fundamental frequency. Increasing the number of stages, i.e., the length of the sequence, the lines in the spectrum become denser and at the same time the power of each is reduced by an inverse proportionality to the increased period. The last term in Equation (3.31) is representative of dc power since in each period the waveform has one polarity of $\frac{(p+1)}{2}$ times and $\frac{(p-1)}{2}$ of the other polarity. The dc component is $\pm\frac{1}{p}$ and the dc power is $\frac{1}{p^2}$.

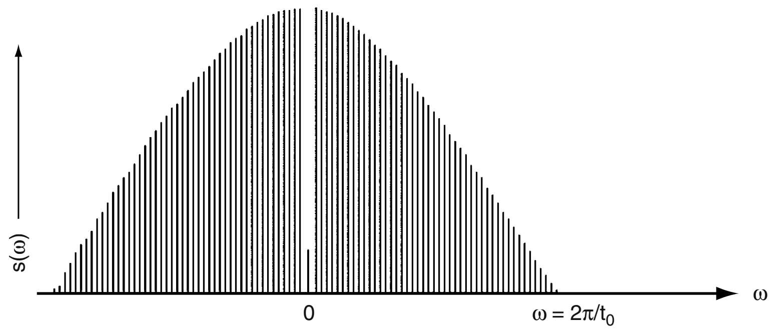


Figure 3.20: Auto-correlation function of maximum length linear shift register sequence.

This PRBS might be converted into a noise like signal by applying it to a low-pass filter. The structure of the codes, however, suggests another more elegant method of conversion. Because all possible combinations of ones and zeros, except all bits zero, occur once in a cycle in the contents of the shift register, the number of ones and zeros will have a binomial probability distribution. That is, individual contents of the different stages in the shift register follow a binomial distribution. When the contents of the shift registers are added up, the distribution tends to be Gaussian (c.f., Central Limit Theorem). The pseudorandom noise (PRN) is generated by taking the weighted average of the LFSR, instant by instant. In this case by adding (equal by weighting) the contents of the shift register at each clock by means of a network of N equal resistors, we get a PRN sequence. Figure 3.21 shows the random noise generated by this technique.

This random noise is not strictly "white". The ACF has a $R_{xx}(0)$ which is σ^2 times the $R_{xx}(0)$ of the PRBS, where σ^2 is the variance of the amplitude distribution and has finite positive values till (*n-1*) shifts of the clock. Beyond that it assumes a steady negative value till it nears the end of the period, where it again peaks up and mirror images the R_{xx} values at the start of the period. The corresponding PSD has a sinc function character with main and secondary lobes and a small dc content.

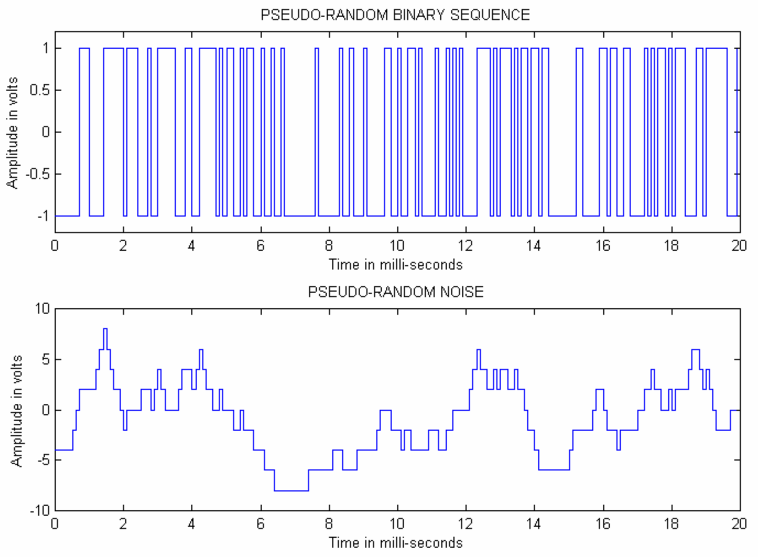


Figure 3.21: Pseudo-random noise generated by the maximal length linear shift register scheme.

The initial correlation and the secondary lobes are not of much consequence as with increasing period $p = (2^n - 1)$ of the sequence which depends on n, the number of stages of the shift register, PRN nearly approximates white noise having Dirac δ-function as the ACF and a fairly steady value for the power spectrum.

If one generates a PRBS which is 1023 bits long (such a PRBS can be generated from a 10-bit long shift register with modulo-2 addition of the contents of 7th and 10th registers fed back to the first stage), one could generate a PRN sequence 1023 long. Leaving the first and the last few sequence members which are correlated among themselves (upto 9 shifts), one can easily generate a multiple valued random sequence of length 1000. The ACF and PSD of such a PRN sequence, generated by the discrete Fourier transform (DFT) and inverse DFT method, is given in Figure 3.22 and those obtained by using the MATLAB® inbuilt functions are given in Figure 3.23.

The advantages of using the pseudo-random noise sequence are:

1 The noise is generated by a deterministic device, it could easily be generated by a feedback shift register and which in turn could be part of an IC, yet it passes the randomness tests [35].

2 This noise signal is multiple valued and not bi-valued only as in PRBS. The noise amplitude could be easily adjusted covering linear and/or non-linear range. The amplitude distribution is near Gaussian.

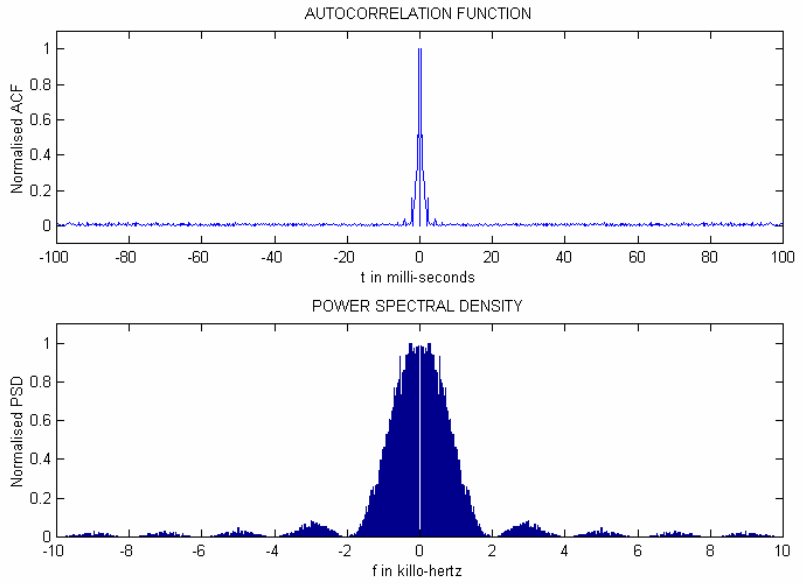


Figure 3.22: Auto-correlation function and the power spectral density of the pseudo-random noise generated with a 10-bit long shift register.

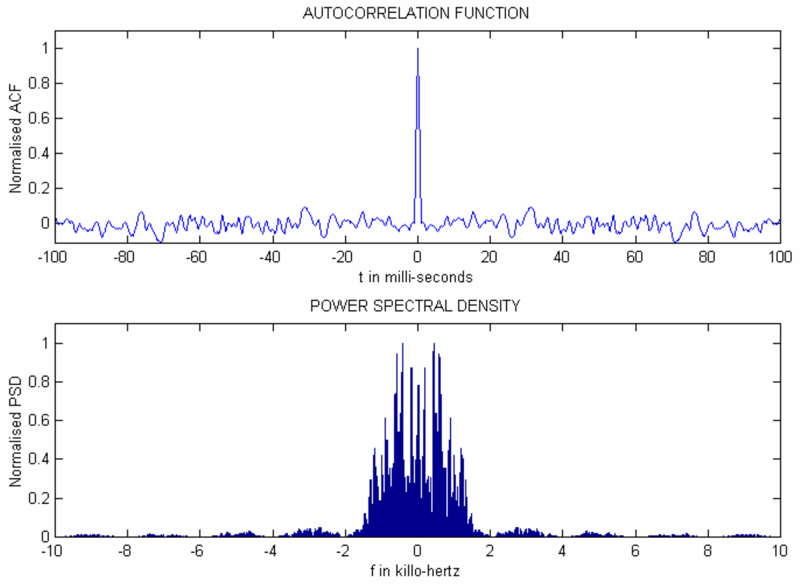


***Figure 3.23:** Auto-correlation function and the power spectral density of the pseudo-random noise generated with a 10-bit long shift register generated by the MATLAB® in-built functions.*

3 The ACF and PSD of the PRN signal could be made to approximate white noise signal. Additionally, it has a small dc component. The period of the sequence should be large enough compared to the time constant of the tested circuit and thus ensuring that nearly all frequency components of interest are retained in a near flat spectrum in the excitation signal.

4 Also, this is the best approximation to a 'persistently' exciting signal which is important for identification experiments [36].

Summing up, the clock frequency of the LFSR and the number of stages has a strong effect on the spectral content of the PRN test stimulus. However, the PRN has more or less constant frequency magnitude over a frequency range of interest. The advantages of the PRN as the test signal are that it will excite the CUT with large number of frequencies simultaneously with varying amplitude levels and will thus allow fault diagnosis to be performed through outputs of the CUT.

3.6 Summary

This chapter has tried to give the reader the idea of different test stimulus generators. Both the deterministic and pseudo-random test pattern generation has been discussed. Section 3.2 described the conventional analog test stimulus generation. The basic sinusoidal oscillators that can be used as test pattern generator was presented in sub-section 3.2.1. Digital test stimulus generation was presented in section 3.3. Section 3.4 covers the Delta sigma signal generation. Pseudorandom noise generation that was *actually* used as test signal was discussed in section 3.5. The circuit for realizing maximum length linear feedback register was introduced. The general principle of the circuit was also covered. After studying this chapter, the reader should be able to justify the use of a particular test signal. The subject matter presented in this chapter will be applied in the fault diagnosis of analog integrated circuits and the next chapter will provide the technique of fault detection and diagnosis in some typical analog circuits.

Exercises

3.1 'An oscillator uses positive feedback.' – do you agree? What are the conditions for oscillation? What are the corresponding gain and phase margins? What is required in a *practical* oscillator to start the oscillations and what limits the amplitude of oscillations?

3.2 Where is the (four arm) bridge in the Wien bridge oscillator (Fig. 3.2 in the text)? What voltage is *input* to the bridge and to where is connected the bridge *output*? There is a series R-C circuit and a parallel R-C circuit. How do they determine the (unique) frequency of oscillation of the bridge?

3.3 We spoke about a Colpitts oscillator but not a Hartley oscillator - why? Even the inductance in the Colpitts oscillator circuit is a *strictly no, no*. What artifice is used instead in the integrated circuit realization of the oscillator? Explain.

3.4 The *IRE Trans. Space, Electron, Telem., Set-8, 204-209 (1962) Paper by H Inose, Y Yasuda and J Murakami* first speaks of a telemetering system by code modulation - Δ-Σ modulation. What precise pulse code modulation system is the precursor of Δ-Σ modulation? An adaptive step size change is required to get around the difficulty of tracking a fast changing signal or to avoid oscillations in this kind of modulation. What are these two difficulties known as? Would they affect Δ-Σ modulation also?

3.5 What is a Δ-Σ attenuator? Two digital integrators, a low pass Δ-Σ modulator, a multiplexer and a low pass filter acting on the single bit stream (Fig. 3.11 of the text) give rise to a very-high-quality low frequency analogue sine wave. Explain the contribution of each and every component in realizing the same, from first principles.

3.6 Hence explain the working of the circuit in Fig. 3.16 in the text.

3.7 A 1023 bit long pseudo-random binary sequence (PRBS) could be generated from a 10-stage long shift register with modulo 2 additions of the contents of the 7th and 10th register fed back to the first stage. Write a computer program to generate the sequence, both uni-polar and bi-polar. Plot the PRBS and the pseudo-random noise (PRN) generated by the above sequence (cf., Fig. 3.21 of the text). [Note: In a bi-polar sequence, zeros are replaced by minus ones.]

3.8 Golomb (1967) (cf., Ref. [35]) specified tests for pseudo-randomness of a PRBS. They are 'balance property', 'run property' and the 'correlation property' respectively of the sequence. Apply the tests to the above PRBS generated and check if they pass the above tests.

[Balance property: Number of ones differs from the number of zeros by at most one.

Run property: Among the runs of ones and zeros in each period (a pseudo- random sequence is never strictly *a periodic* due to the deterministic generation device), one half of the runs of each kind are of length one, one fourth of each kind are of length two, one eighth are of length three, and *so on* as long as these fractions give meaningful number of runs.

Correlation property: If a period of the sequence is compared, term by term, with any cyclic shift of itself, the number of agreements differs from the number of disagreements by at most 1.]

3.9 Do the amplitude distribution of the PRN generated above, follows an *approximate* Gaussian distribution, giving rise to a pseudo-random noise with Gaussian amplitude density distribution function (PRNG)? Plot and see.

3.10 By using computer programs, generate the auto-correlation function (ACF) and power spectral density (PSD) of the uni-polar and bi-polar PRBS and the corresponding PRNG sequences (cf., Figs. 3.22 and 3.23 of the text).

References

[1] B. Davis, The Economics of Automatic Testing, McGraw-Hill, London, 1982.

[2] S. D. Millman, "Improving quality: yield versus test coverage", Journal of Electronic Testing: Theory and Applications, Vol. 5, pp. 253-261, 1994.

[3] B. Davis, "Economic modeling of board test strategies", Journal of Electronic Testing: Theory and Applications, Vol. 5, pp. 157-170, 1994.

[4] V. Agrawal, C. Kime, K. Saluja, "A Tutorial on Built-In Self-Test", IEEE Design & Test of Computers, pp. 73-80, March 1993, and pp. 69-77, June 1993.

[5] E. McCluskey, "Built-In Self-Test Techniques", IEEE Design & Test of Computers, pp. 21-36, April 1985.

[6] J. Abraham, "Fault Modeling in VLSI", in Advances in CAD for VLSI, Vol. 5: VLSI Testing, T. Williams (ed.), Elsevier Science Publishers, Amsterdam, 1986.

[7] P. Bottorff, “Test Generation and Fault Simulation”, in Advances in CAD for VLSI, Vol. 5: VLSI Testing, T. Williams (ed.), Elsevier Science Publishers, Amsterdam, 1986.

[8] M. Sheu, C. Lee, “Simplifying Sequential Circuit Test Generation”, IEEE Design & Test of Computers, pp. 29-38, Fall 1994.

[9] I. Pomeranz, S. Reddy, “LOCSTEP: A Logic Simulation Based Test Generation Procedure”, Proc. 25th International Symposium on Fault-Tolerant Computing (FTCS-25), Pasadena, CA, pp. 110-119, June 27-30, 1995.

[10] T. Fujino, H. Fujiwara, “An Efficient Test Generation Algorithm Based on Search State Dominance”, Proc. 22nd International Symposium on Fault-Tolerant Computing (FTCS-22), Boston, Mass., pp. 246-253, July 8-10, 1992.

[11] J. Roth, W. Bouricius, P. Schneider, “Programmed Algorithms to Compute Tests to Detect and Distinguish Between Failures in Logic Circuits”, IEEE Trans. Electronic Computers, Vol. EC-16, pp. 547-580, 1967.

[12] P. Bardell, W. McAnney, J. Savir, Built-in Test for VLSI, Pseudorandom Techniques, John Wiley & Sons, New York, 1987.

[13] B. Murray, J. Hayes, “Testing ICs: Getting to the Core of the Problem”, IEEE Trans. Computers, Vol. 29, No. 11, pp. 32-38, 1996.

[14] A. Fuentes, R. David, B. Courtois, “Random Testing versus Deterministic Testing of RAMs”, Proc. 16th International Symposium on Fault-Tolerant Computing (FTCS-16), Vienna, Austria, pp. 266-271, July 1-4, 1986.

[15] E. Eichelberger et al., Structured Logic Testing, Prentice Hall, Englewood Cliffs, N.J., 1991.

[16] A.S. Sedra and K.C. Smith, Microelectronic Circuits, 4th Edition, Oxford University Press, New York, 1998.

[17] J. Tierney, C.M. Radar and B. Gold, “A digital frequency synthesizer”, IEEE Trans. Audio Electroacoust., Vol. AU-19, pp. 48-57, 1971.

[18] V. Manassewitch, Frequency Synthesizers, Theory and design, 2nd Edn., John Wiley & Sons, New York, 1989.

[19] Stanford Telecom, Direct Digital Synthesizer Handbook, Santa Clara, CA, 1990.

[20] H.T. Nicholas and H. Samueli, “A 150MHz direct digital frequency synthesizer in 1.25um CMOS with -90dBc spurious performance”, IEEE Jour. of Solid-State Circuits, Vol. 26, pp. 1959-1969, 1991.

[21] L.T. Bruton, “Low sensitivity digital ladder filters”, IEEE Trans. Circuits and Syst., Vol. CAS-22, pp. 168-176, 1975.

[22] M.S. Lee and C. Chang, “Switched capacitor filters using the LDI and bilinear transformations”, IEEE Trans. Circuits and Syst., Vol. CAS-28, pp. 265-170, 1981.

[23] L.E. Turner and B.K. Ramesh, “Low sensitivity digital LDI ladder filters with elliptic magnitude response”, IEEE Trans. Circuits and Syst., Vol. CAS-33, pp. 697-706, 1986.

[24] L.E. Turner, “A fully programmable digital oscillator”, CMC Workshop, Kingstone, June 1992.

[25] M.W. Hauser, “Principles of oversampling A/D conversion”, Journal of Audio Engineering Society, Vol. 39, No. 1-2, pp. 3-26, Jan – Feb 1991.

[26] D.A. Johns and D.M. Lewis, “Design and analysis of delta-sigma based IIR filters”, IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. CAS-40, No. 4, pp. 233-240, April 1993.

[27] X. Haurie and G.W. Roberts, “Arbitrary-precision signal generation for bandlimited mixed-signal testing”, Proc. 1995 IEEE International Test Conference, Washington, pp. 78-86, October 1995.

[28] A.K. Lu, G.W. Roberts and D.A. Johns, “A high-quality analog oscillator using oversampling D/A conversion techniques”, IEEE Trans. Circuits and Systems-II, Vol. 41, No. 7, pp. 437-444, July 1994.

[29] J.C. Candy, “A use of double integration in sigma-delta modulation”, IEEE Trans. Communications, Vol. 33, No. 3, pp. 249-258, Mar. 1985.

[30] R. Schreier and M. Snelgrove, “Band pass sigma-delta modulation”, Electronics Letters, Vol. 25, No. 23, pp. 1560-1561, Nov. 1989.

[31] B.R. Veillette and G.W. Roberts, “Delta-sigma oscillators: versatile building blocks”, International Jour. of Circuit Theo. and Appl., Vol. 25, pp. 407-418, 1997.

[32] A.K. Lu and G.W. Roberts, “An analog multi-tone signal generator for built-in self-test applications,” Proc. 1994 IEEE International Test Conference, Washington, pp. 650-659, Oct. 1994.

[33] H.H. Schreiber, “Fault dictionary based upon stimulus design”, IEEE Transactions on Circuits and Systems, Vol. CAS-26, pp. 529-537, September 1979.

[34] S. Sinha and B.V. Raja Rao, “A pseudo random noise generator”, Institution of Engineers (India) Journal - E.T., Vol. 50, pp. 143-146, May 1970.

[35] S.W. Golomb, Shift Register Sequences, Holden-Day, San Francisco, 1967.

[36] K.J. Astrom and P. Eykhoff, “System identification - a survey”, Automatica, Vol. 7, pp. 123-162, March 1971.

Chapter 4

FAULT DIAGNOSIS METHODOLOGY

4.1 Introduction

Majority of the analog circuit testing techniques can be classified as either specification-driven or fault-model-driven. In the specification-driven techniques, the correctness of the circuit under test (CUT) is verified by checking a set of measured parameters with the associated tolerance ranges. The CUT is declared as faulty if any of the measured parameters falls outside the tolerance range. In the fault-model-driven techniques, a fault list is first constructed and the fault is declared as detected if the discrepancy between the output of the nominal design and the faulty circuit is greater than certain derived range, which limits come from measurement error, process variation, etc. Therefore, for fault-model-driven techniques, the observables for the fault detection should be selected carefully such that the deviation caused by the defect can be easily observed and measured. A fault can be either catastrophic or of parametric kind. The former is caused by physical defects, which change the topology of the CUT (e.g., dead short/bridging faults) while the latter is caused by the variations in the device parameters. In general, the CUT subject to catastrophic faults will be completely out of function and therefore these faults are easily detectable. On the other hand, the circuits subject to parametric variation will still be functioning but there is performance degradation.

As analog circuit faults are continuous in mode they can take an infinite number of values and so, the only difference between catastrophic and parametric faults depend on the concept of "totally malfunctioning circuit". Also, faults considered catastrophic at one description level may become parametric at a higher one. Further, a good knowledge of the probability of occurrence of all possible defects is necessary for actual fault coverage by a test methodology. To locate which defects are more likely to occur requires the gathering of a high volume of production and field failure data. Considerable time is needed for such measurements and analysis. It is reported in the literature [1] that 80-to-90 percent of analog faults involve shorted and open resistors, capacitors, diodes and transistors. So to verify the efficiency of a test method, the fault models of section 2.3 can be used. Ohletz [2] presents the following reasons for this choice:

- The hard faults are mainly responsible for failures of the circuit to operate.
- A test method that detects 100% of hard faults may also detect most of the soft faults.

The efficiency of any fault-model-driven testing technique is primarily determined by the fault detection capability of the selected observable output and the cost for generating and measuring such observable output. Any DC or AC parameter can be used as observable output for fault detection. The fault detection capability in case of AC parameters may be better than that of DC parameters.

4.2 Fault diagnosis procedure

The general idea is to test the response of the circuit at a given frequency. Deviations in circuit parameters caused by any fault will affect the output response, either in its amplitude or phase. The test engineer must supply a minimum number of excitation frequencies in order to detect various component faults.

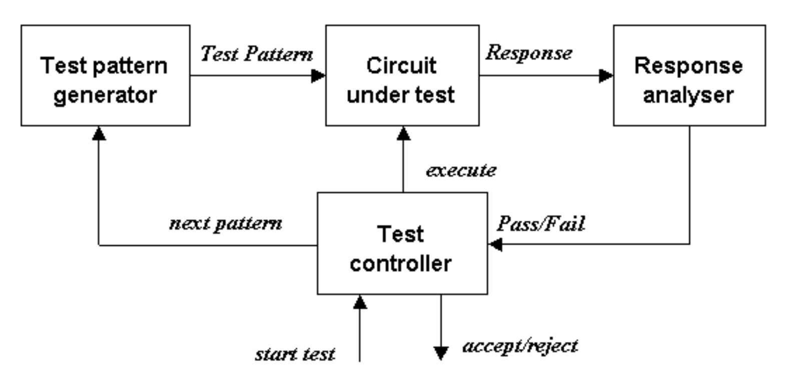


Figure 4.1: Architecture of a typical fault diagnostic set-up.

As shown in Figure 4.1 the test of an electronic circuit is a stimulus/response measurement. In the first step a test pattern is applied to the circuit to bring it to a defined initial state or exercise some functionality.

In the second step the circuit processes the test pattern and in the third step the circuit's response is checked. This test procedure is repeated for different test patterns by a test controller. Basically a test signal is applied to the circuit under test (CUT) and the response of the system is analysed. The analyser matches the response or responses to a pre-existing knowledge base and classifies the behaviour of the CUT.

A precision analog test stimulus generator is an essential component of the fault detection and diagnosis scheme of analog integrated circuits. One problem central to testing is the determination of an optimal test pattern fulfilling the following essential requirements:

– detection of (ideally) all defects assumed in the fault model,
– ease of generation/storage (low overhead) and
– compactness (short test stimulus generation time).

This stimulus generator depends on the type of test measurement to apply. Four categories of measurements can be distinguished:

1. *DC static measurements:* They include the determination of the DC operating points, DC bias and DC offset voltages and DC gains. DC faults are detected by a single set of steady state inputs.
2. *AC dynamic measurements:* They measure the frequency response of the circuit under test. The input stimulus is usually a sine waveform with variable frequency. Harmonic spectral analysis can be performed using DSP techniques.
3. *Time domain measurements:* They use pulse signals (square wave, step and pulse trains), ramps or triangular waveforms as the input stimuli of the circuit. Some of the parameters derived are slew rate, rise and delay times.
4. *Noise measurements*: They measure the variation in the signal that appears at the circuit's output when the input is set to zero.

Depending on the circuit, some or all of these measurements may be required. For any such measurement a basic component of the diagnostic setup is the test pattern generator details of which have been discussed in Chapter 3. The output response of the CUT contains all the necessary diagnostic information. This information can now be processed (or analysed) in a number of ways to extract measures of various parameters. Those will provide the desired diagnosis of the CUT.

4.3 Fault dictionary techniques

The fault dictionary approach has been found very useful for diagnosis of hard faults (open and short circuits) in analog circuits. The fault dictionary method is actually dated back long before the advent of digital computers. For example, service manuals accompanying electronic equipments usually contain a section on ‘trouble-shooting’, where a step-by-step testing procedure is described, together with a table of symptoms and possible causes. That is in essence a fault dictionary approach. One can compile a fault dictionary manually, but with the use of digital computers the whole process can be automated. As a result, the fault dictionary can be made much more comprehensive and easier to use.

It is well known from literature that the fault dictionary approach is incapable of diagnosing soft failures, i.e., component parameter value drift. Still the fault dictionary approach will remain a very valuable part of an overall fault diagnosis scheme. This is based on the following observations:

1. Statistics show that hard faults account for about 80-90 percent of faults in analog circuits [3].
2. The fault dictionary approach is well suited for diagnosing hard faults; although the initial effort to compile the fault dictionary is quite demanding, that task is done once and for all.
3. A test method that can detect 100 percent of the hard faults may also detect most of the soft faults [2].

Summarising, the basic analog tests consist of exciting the circuit-under-test (CUT) and obtaining the response. Depending on the purpose of the test, the excitation may be a dc, sinusoid, square-wave, or some arbitrary waveform. Which excitation is to be used depends on the type of measurement that is to be taken. There are four main measurement categories: *DC measurements, AC measurements, Transient or time-domain measurements and Noise measurements.*

Usually the applied inputs and the measured outputs in a fault dictionary can either be dc or ac depending on the type of measurements carried out. But the dc approach has the obvious advantages of simpler circuit simulations and easier output measurements. Therefore, the present discussion shall be confined to the dc fault dictionary. The dc fault dictionary approach consists of two distinct stages:

- *Stage 1*: Pre-test analysis to compile the fault dictionary. In this stage the analog CUT is simulated by a digital computer program under nominal (fault-free) as well as all pre-selected hard faults. Judiciously chosen dc input voltages are applied. The induced dc voltages at a selected set of test nodes are calculated and stored to constitute the fault dictionary.
- *Stage 2*: Post-test analysis to identify the faults. In this stage, measurements of test node voltages are made on the actual CUT and the measured values are compared with those stored in the fault dictionary. First, a fault detection algorithm is applied to determine whether the circuit is faulty at all. If the answer is affirmative, then

the fault is identified by the application of some fault isolation algorithm.

4.3.1 Pre-test analysis

In the very first step, the test engineer must come up with a reasonable fault list, based on his experience and past history of failures of the CUT. In the next step, the test nodes are to be selected from among the accessible nodes. One may initially choose all or most of the accessible nodes as the test nodes in the absence of any guideline. After the simulation of the CUT under all fault conditions have been carried out, a suitable fault isolation algorithm may be devised to reduce the number of test nodes and yet maintain satisfactory fault isolation.

In general, the fault-free and the faulty CUT must be simulated under more than one input combination in order to achieve adequate isolation of faults. Each input combination is also called an input vector. So far no algorithm has been developed for the selection of input vectors for analog circuits. Trial and error and the knowledge of the particular CUT seems to be the only avenue. Selection of the test vector or test stimuli is initially made by the test engineer and expanded or modified if detection (separation from the nominal case) and isolation (separation among faulty cases) requirements are not met. More detail about the test stimulus generation has already been described in Chapter 3.

After the selection of the fault list, test nodes and input vectors the CUT must be simulated for one fault-free and all the faulty conditions of the CUT presenting one set of input vectors at a time and gather all the test node voltage data from these simulations. These data will later be processed by the fault isolation algorithm to reduce the number of test nodes. The data corresponding to the retained test nodes will then constitute the fault dictionary.

Before proceeding further, it is important to introduce the concept of ambiguity. Let us consider a hypothetical case of a given circuit with two test nodes designated as TP1 and TP2, and eight faults designated as F1 to F8. The nominal or fault-free condition of the circuit is designated as F0. The test node voltages have been determined using a computer-aided simulation program for the circuit under the fault-free and the eight fault conditions yielding the results shown in Table 4-I.

All these voltage values are obtained under the assumption of exact values of the components used in the circuit. As explained in section 2.3, the value of any component may vary within some tolerance range. If the measured value of TP1 voltage is say 5.1 V, one really cannot be certain whether the circuit is under F0 or F8 condition. All one can say is that when

TP1 voltage is 5.1 V, the circuit is under either F0 or F8 condition. Thus in this case, F0 and F8 form what is called an ambiguity set. One may now define the voltage of an ambiguity set to have a range about its centre value and that the different ambiguity set voltages do not overlap. In the above example, test node 1 has 3 ambiguity sets and test node 2 has 4 ambiguity sets which are listed in Table 4-II below.

Table 4-I: ***Simulated test node voltages of the hypothetical circuit for fault-free and various fault conditions.***

	F0	F1	F2	F3	F4	F5	F6	F7	F8
TP1 voltage	5.0	7.0	7.4	7.3	7.2	9.6	9.7	9.8	5.2
TP2 voltage	9.0	5.0	7.1	6.4	6.2	5.1	5.2	7.3	9.2

Table 4-II: ***ambiguity sets' contents and voltage ranges.***

Node, ambiguity set	Circuit condition	Node voltage range
(1,1)	F0,F8	4.6 – 5.6
(1,2)	F1,F2,F3,F4	6.7 – 7.7
(1,3)	F5,F6,F7	9.2 – 10.2
(2,1)	F1,F5,F6	4.6 – 5.6
(2,2)	F3,F4	5.8 – 6.7
(2,30	F2,F7	6.6 – 7.7
(2,4)	F0,F8	8.6 – 9.6

From Table 4-II one can conclude that fault F1 cannot be isolated from faults F2, F3 and F4 if test node 1 only is used. Similarly fault F1 cannot be isolated from faults F5 and F6 if test node 2 only is used. However, if both the test nodes are used then fault F1 can be isolated. On the other hand faults F3 and F4 cannot be isolated even if both the test nodes are used.

In order to obtain maximum isolation, the ambiguity sets must be manipulated to determine which faults can be isolated and which nodes will provide the greatest degree of isolation. Hochwald and Bastian [3] have suggested the following rules to uniquely identify a fault:

Rule 1. Any ambiguity set which has a single fault within it uniquely defines that fault at that test node.

Rule 2. Ambiguity sets whose intersection or symmetric difference result in a single fault also uniquely defines the fault.

4.3.2 Post-test analysis

After a circuit (fault-free or faulty) has been tested and all the measured test node voltages are collected one should use some post-test algorithm to isolate a fault. One method is the calculation of the sum of square deviations corresponding to a particular fault. If $SSD(F_J)$ denote the sum of square deviations corresponding to the fault F_J then we can write

$$SSD(F_J) \overset{\Delta}{=} \Sigma(\text{measured node voltage} - \text{calculated node voltage})^2 \quad (4.1)$$

where the summation is over all test nodes and all input vectors for the fault F_J. Then after $SSD(F_J)$ have been calculated for all faults, the one leading to the smallest value is considered to be the fault that has occurred.

4.4 DSP based techniques

In a production environment, analog and mixed-signal circuits have long been tested using the digital signal processing (DSP) based testing [4]. In the DSP-based testing, the desired test stimulus is generated in digital form and converted to an analog signal by passing it through a digital-to-analog (D/A) converter. The resulting analog signal is then applied to the circuit-under-test (CUT) and the CUT output response is digitised by an analog-to-digital (A/D) converter. A DSP system is then used for performing further manipulations. Depending on the type of measurements required appropriate software programs are used in the DSP to emulate the function of analog instruments. Figure 4.2 illustrates the basic DSP based test system.

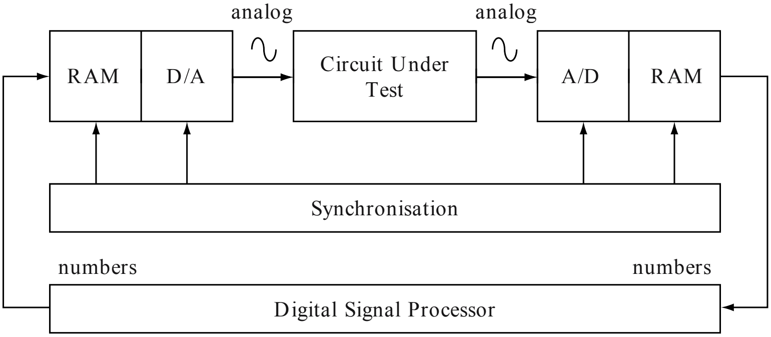


Figure 4.2: DSP based test system.

As can be seen from Figure 4.2 the synchronisation of the signal generation with the digitisation process is essential for proper operation of such a system [4]. It is observed that the DSP-based testing methods are efficient and the same set up can be used to test various analog and mixed-signal circuits using functional testing [5] or pseudo-random testing [6][7]. The first obvious advantage that comes from this approach is the flexibility that programmability provides. The same hardware can be used to perform a multitude of test functions. Secondly, the correction factors associated with a system calibration can be easily incorporated into the routine of any emulated instrument, thus correcting for the effects of drift and aging that comes from running test equipment continuously for 24 hours a day, 7 days a week. A less obvious advantage, but equally important, is the ability to pipeline the different phases of the test procedure. On account of the discrete events that are taking place, operations that do not need to run in real-time can be delayed and run in parallel with the operations of the next device. Finally, *coherent testing* provides a means in which to gather information using the least number of samples. Owing to its importance to DSP-based testing, a brief description is given below.

One may be wondering at this point whether digitising the excitation and response of the CUT somehow degrades the accuracy of the measurement or loses information. The answer is simply no, as Shannon so succinctly pointed out [8]. His observation suggests that the DSP-based test station has the same amount of information to work with as does the conventional test setup shown in Figure 4.1. Mahoney [9] pointed out that a DSP-based test station using an N-point Fast Fourier Transform (FFT) with rectangular windowing (i.e., no additional post-processing) can perform a much faster and more accurate frequency-selective power measurement than the setup of Figure 4.1. This is a result of coherent sampling and by eliminating the analog implementation of the squaring operation in the true-RMS power meter. He coined this test arrangement as *coherent testing*. As test time is of primary importance in a production environment, this observation was significant and lead to the creation of a new family of mixed-signal testers. Basically, coherent testing establishes which input test frequencies can be used to perform an accurate sinusoidal test using an N-point FFT. According to [4], the test frequency f_T is selected according to the following

$$f_T = \frac{M}{N} f_S \tag{4.2}$$

where f_S is the sampling rate and M represents an arbitrary integer, usually less than N/2 and has no factors common with N, i.e., M and N are relatively-prime. The latter ensures that each point is unique in the N-point set, thereby maximizing the information content. An alternative description of the relationship given in Equation (4.2) is that the test frequency should only be selected as a harmonic of f_S / N, the so-called primitive frequency.

Present analog circuit testing relies heavily on the DSP based testers because they overcome the limitations of purely analog testers and provide the following advantages:

- *Accuracy:* The DSP-based testers are more accurate than analog only tester, because a DSP avoids various errors and information losses due to unwanted analog components. Crosstalk, noise, signal drift and nonlinearity in analog components are significantly reduced in the DSP-based testers.
- *Speed:* The DSP-based testers are faster and can produce many different emulated analog test instrument measurements simultaneously using a discrete Fourier transform (DFT) or fast Fourier transform (FFT). For multiple measurements, a DSP-based tester is more efficient than the use of multiple analog test instruments.
- *Ease of operation:* The DSP-based testers are simple to calibrate and testing is more repeatable. For example, software built into the tester can store offsets and gain adjustment factors and can apply them automatically when measurements begin. Also, a DSP-based tester requires reduced maintenance because there are fewer continuous variable electrical adjustments to be done by a technician.
- *Flexibility in testing*: The DSP-based testers are more flexible. Test conditions can easily be changed by the operator from a computer terminal rather than adjusting many analog controls in the analog testers. Also, the DSP-based testers can easily model a fault-free and faulty device/circuit compared to its analog counterpart.
- *Measurement information:* The DSP-based testers can provide more information along with the desired test parameters because such information can easily be obtained by slight modification in the built-in software.
- *Size and Power:* A general purpose DSP-based tester is smaller in size, cheaper and usually consumes less power compared to its analog counterpart.

However the DSP-based testers have the following limitations:

1. When making only one measurement a conventional analog tester is much cheaper.
2. The operator has to know the instrument theory to operate the DSP-based testers because of their sophistication in design.

3. The test engineer must have prior knowledge of the physical and mathematical principles of each test.
4. Last but not the least, the DSP-based testers are expensive although their cost is coming down due to availability of low cost VLSI technology.

4.5 Model based observer technique

This section outlines the basic concepts of analog circuit fault diagnosis using model-based observer scheme. The concept of model-based observer scheme has been well explained in the book edited by Patton, Frank and Clark [10]. The crux of any model-based fault detection and identification (FDI) scheme is the influence of un-modelled disturbances as, for example, parameter uncertainties, changes in the system parameters, and system and measurement noise. The effect of these obscures the performance of the FDI system and act as a source of false alarm. Therefore, in order to minimise the false alarm rate, the observer be so designed that it becomes robust to these uncertainties. The concept of model-based FDI in dynamic process plants is well known [11][12] and the basic principle is illustrated in Figure 4.3.

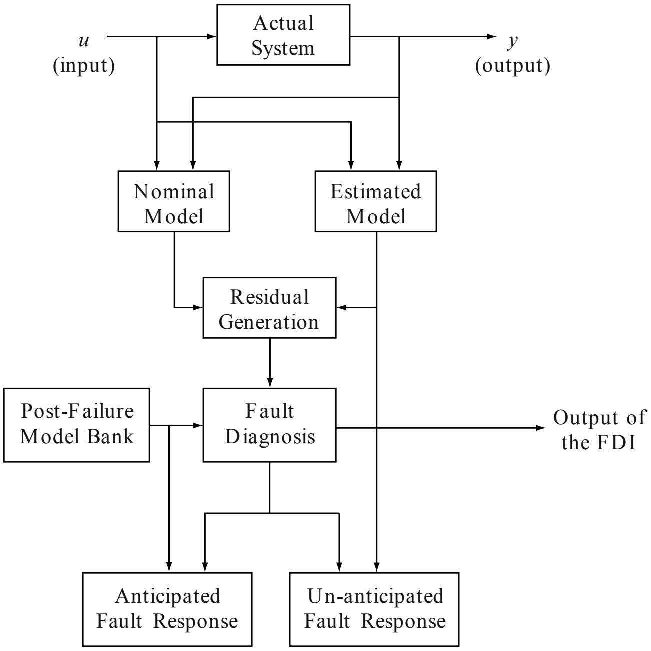


Figure 4.3: ***Basic principle of observer based fault detection and isolation.***

The general procedure of observer-based fault detection and diagnosis consists of three steps:

1. generation of residuals, i.e., of functions carrying information about the faults,
2. decision on the occurrence of a fault and
3. isolation of the faulty component, i.e., localisation of the fault.

The estimated model is a continuously updated mathematical representation of the physical system. To detect any changes in the system dynamics, the estimated model is compared to a nominal model. Residual generation gives a measure of the deviation between the estimated model and the nominal model. This measure can be expressed in terms of the system outputs, the state variables, some critical system parameters, or more generally, a combination of the above. Based on the residual vector, a decision is made on whether a failure has occurred. If a failure is detected, a parallel decision mechanism compares the characteristics of the failure (as provided by the residual vector) with the signatures of known failure modes. Pattern recognition techniques and associative memories provide effective means for achieving this task.

Signatures of known failure modes, which are referred to as anticipated fault responses, are stored in a post-failure model bank. Depending on the specific application, the number of distinct signatures stored in the post-failure model bank can range from one to hundreds. Making provisions for common failure modes gives the advantage of a fast, predetermined response to such failures. Therefore, it is important that the post-failure model bank be updated periodically to include the signatures of recurring (and well understood) failures. In the case of a failure being diagnosed, an output signal is transmitted to a desired monitor interface which appraises an external monitoring system about the failure. In many cases unexpected failures are encountered. This may occur as a result of minimal knowledge about the things that can go wrong in a specific system, due to unexpected breakdowns, or even as a result of errors in preparing the post-failure model bank. In the case of a failure that does not match any of the anticipated failure modes, the unanticipated failure mode is set to trigger an output as such. This may form part of a 'learning process' for the FDI system.

The fault isolation task may require different observer configurations for locating the fault. The basic principle of observer-based residual generation is as follows:

A set of measured variables, $\boldsymbol{y}_i$, of the actual system is compared with the corresponding signals of the nominal model, $\hat{\boldsymbol{y}}_i$. The difference $\boldsymbol{e}_i = \boldsymbol{y}_i - \hat{\boldsymbol{y}}_i$ represents the observation error or innovation. For the purpose of fault diagnosis, $\boldsymbol{e}_i$, or a function of $\boldsymbol{e}_i$, can be used as a residual, $\boldsymbol{r}_i$, that carries information about the fault. If no fault occurs, the observer will track the system so that $\boldsymbol{r_i}$ depends only on the un-modelled disturbances such as system and measurement noise, acceptable parameter variations or modelling uncertainties. If, however, a fault in the system occurs, the observer models the system with less accuracy and the magnitude of $\boldsymbol{r_i}$ will increase. Hence, the fault can be detected by checking the increase in $\boldsymbol{r_i}$ caused by the fault (or faults).

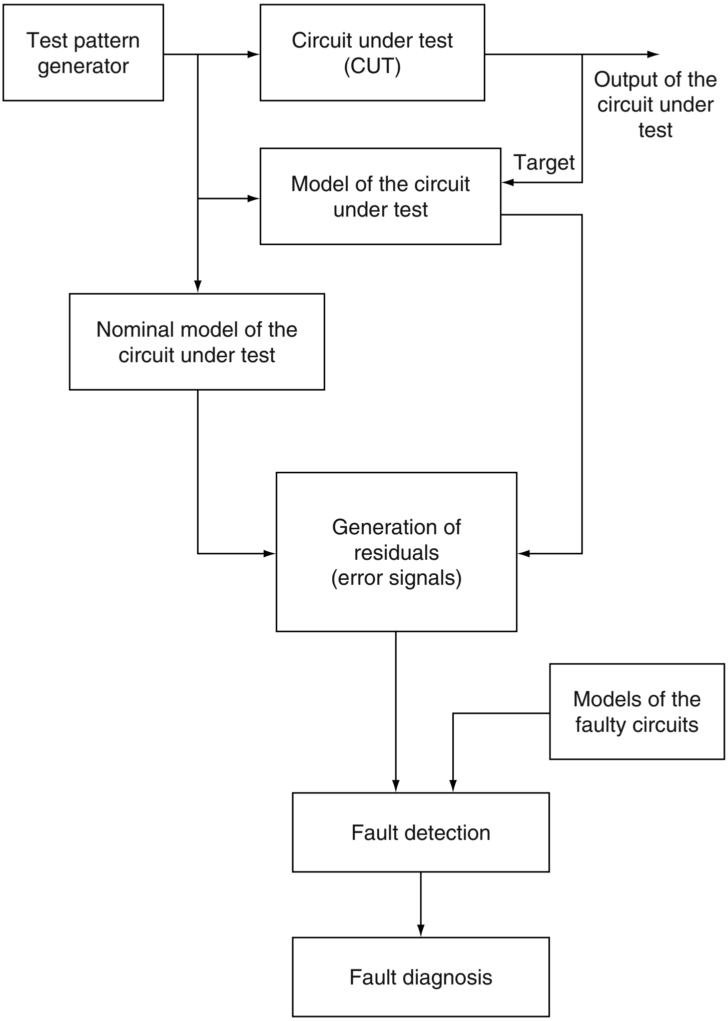

Test pattern generator
Circuit under test (CUT)
Output of the circuit under test
Target
Model of the circuit under test
Nominal model of the circuit under test
Generation of residuals (error signals)
Models of the faulty circuits
Fault detection
Fault diagnosis

Figure 4.4: ***System overview for fault detection and diagnosis of analog integrated circuits.***

The key problem of the observer-based fault detection and diagnosis procedure is the generation and evaluation of a set of residuals that permits not only the detection but also a unique identification of the fault. In general, a bank of observers or an observer scheme, where each observer is made sensitive to a different fault or a group of faults, whilst being insensitive to un-modelled disturbances, noise, modelling uncertainties and parameter variations within permissible limits, can achieve this goal. Figure 4.4 shows the system overview for fault detection and diagnosis of analog integrated circuits using the above general approach [13].

The diagnostic framework consists of the following steps:

1. *Development of a model:* A mathematical model is developed for the CUT based on physical information and data. The input and output variables of the CUT are clearly defined and all the relevant parameters are identified. The model describes the behaviour of the CUT under normal operating conditions.
2. *Generation of an error signal:* The error signal is the difference between the output of the model of the actual CUT and the output of the nominal model of the CUT. Under normal conditions, the error signal is "small" and corresponds to random fluctuations in the output since the model eliminates all the systematic trends. However, under faulty conditions, the error signal deviates widely from its nominal value and contains systematic trends because the model no longer represents the CUT adequately.
3. *Fault detection and diagnosis:* Different types of faults can develop in the CUT. All these faults cause the error signal to deviate from its nominal value. The error signal so generated is now compared with previously constructed fault signatures to detect and diagnose the faults in the CUT.

Following sections illustrate a few examples of fault diagnosis in analog circuits using artificial neural network trained with back error propagation as the observer model and the pseudorandom noise as input stimulus.

4.5.1 Fault diagnosis in stand-alone analog IC

The objective is to provide a mechanism for fault detection and identification down to the component level. The procedure of the model-based fault diagnosis (based on the general procedure of observer-based fault diagnosis described above) in analog ICs can roughly be divided into the following three stages:

1. fault signature generation,
2. generation of the residuals, and
3. detection and classification of the fault.

The first stage involves the definition of different types of faults. The CUT is simulated using the test stimulus for each type of fault induced. The test stimulus in this case is pseudo-random noise (PRN). Such a signal is a kind of spread spectrum test signal. It is known that for linear time invariant (LTI) systems, the impulse response determines the system properties for a given initial state. The Fourier transform of the impulse response is also the transfer function of the system. So monitoring the impulse response in the time domain will detect changes in the system properties like faults. The consideration that the CUT may be driven outside its region of linear operation, for which it is to be tested, motivates the use of a wide-ranging noise with constant power spectral density over all the frequencies. The output response of the CUT forms the signature for that fault. Figure 4.5 shows the flow diagram to generate these signatures.

All the simulations are done using PSpice® and Monte-Carlo analysis to obtain various faulty as well as fault-free signatures of the CUT. The first run of the Monte-Carlo analysis gives the nominal signature, which is done with the nominal values of all components in the CUT. Subsequent runs are done with values generated randomly by PSpice® within the specified tolerance limit. The maximum value of the output obtained at the end of all the runs gives the greatest difference in the output waveform from its nominal output due to variations in the tolerances of the circuit components. Hereafter this maximum value of the output signature is designated as the CUT output with 'extreme tolerance'.

For the residual generation three kinds of models are required: nominal, actual (observed) and that of the faulty CUT. The achievable quality and high performance of the scheme mainly depends on the quality of the model of the system. It is, therefore, most important to start with a thorough and realistic specification of the given CUT. For a realistic representation, it is important to model all effects that can lead to the detection of a fault, genuine or otherwise. All the models are created using multi-layer artificial neural networks (ANNs) consisting of one input layer, one or more hidden layers and one output layer. These ANN models are trained using the back-error propagation (BEP) algorithm to output the CUT responses under fault-free and faulty conditions. The fault-free ANN model of the CUT with all its components kept at their nominal values represents the nominal model. The difference between the outputs of the nominal model and the model of the actual CUT generates the residual.

When there is no fault in the CUT then the two models (the nominal and the actual) will present a very close behaviour, thereby generating little or no residual. Under faulty condition the CUT will behave differently and so will the ANN model of the CUT. The system will now output a considerable residual showing how much the actual CUT is different from the fault-free

CUT. Any fault in the CUT, be it catastrophic or parametric, will change the residual signature. These residual signatures of known fault conditions of the CUT, known as the fault signatures, are first simulated and stored in a model bank for future comparison.

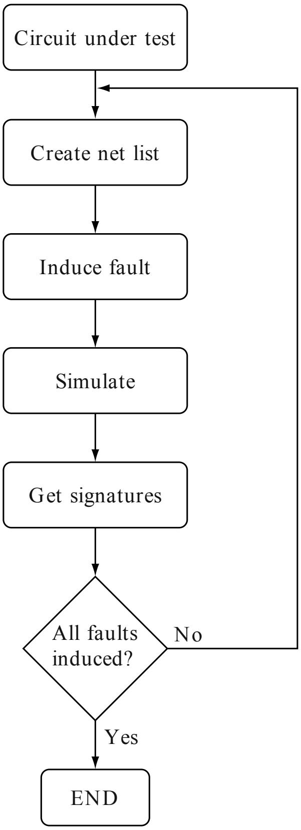


Figure 4.5: Signature generation flow diagram.

The fault diagnosis procedure involves the simulation and modelling of the actual CUT, through the application of the PRN, generating the residual and comparison of this residual to that of different types of residual signatures simulated and stored previously in a model bank as fault signatures. Depending on the specific application, the number of distinct signatures stored in the model bank can range from one to hundreds. Making provisions for common fault signatures gives the advantage of a fast, predetermined response to faults. The resulting fault diagnostic method is quite robust.

Two models are needed for the fault-free as well as for each type of fault. One is the nominal model representing the behaviour of the circuit with all parameters held at their nominal values; the second model represent the behaviour of the circuit for maximum deviation in its output response during the Monte-Carlo simulation of the CUT taking into account extreme variations of the components within their tolerance range. For generation of residuals, these models were excited by the PRN test signal and the output compared to the output of the nominal fault-free model excited by the same PRN. The difference constitutes the error signal, which is the residual for the particular type of fault. All these residuals (the nominal as well as the extreme tolerance) for known fault conditions, known as fault signatures, were stored in a model bank for future reference.

In order to explain the working of the test methodology, the µA 741 Op-Amp is chosen as an example. This is one of the Op-Amp, which is widely used in many analog signal-processing applications. The other example chosen is the MOS Op-Amp from the IEEE Analog and Mixed-Signal Benchmark Circuits [14].

4.5.1.1 Example 1: The µA 741 Op-Amp

The µA 741 Op-Amp is used in the voltage follower configuration. Figure 4.6 shows its schematics. The nominal values of the circuit components are:

$R_1 = 1\text{k}\Omega$, $R_2 = 1\text{k}\Omega$, $R_3 = 50\text{k}\Omega$ $R_4 = 5\text{k}\Omega$, $R_5 = 40\text{k}\Omega$,
$R_6 = 27\Omega$, $R_7 = 22\Omega$, $R_8 = 100\Omega$, $R_9 = 50\text{k}\Omega$, $R_{10} = 50\text{k}\Omega$,
$R_{11} = 50\text{k}\Omega$, $R_{12} = 300\Omega$, $C_1 = 30\text{pF}$.

All the parameters were assigned ±5% tolerance.

The number of fault classes depends on the level of diagnostics desired. Go/no-go testing requires only two classes: faulty and fault-free. For complete component level diagnosis more number of fault classes are required depending on the level of diagnosis. In general possible faults for analog ICs can come from two sources; the first involves the components designed in the IC and the second being the parasitic effects from undesired circuit topologies like parasitic capacitance and bridging faults. By varying the components out of their nominally assigned values within tolerances, the faults associated with the normal circuit topology can be generated. This can also include the short and open circuits. The faults caused by undesired circuit topology can be modelled by placing components such as capacitors and/or resistors between and/or across normal nodes of the circuit. These components can be assigned values between zero and infinity. This defines a systematic method for the required simulation. All faults may not require to be simulated. Circuit performance may be the same for a variety of fault configurations. These types of faults will typically result in the definition of an ambiguous fault group. For the fault simulation the nominal values of the BJT PSpice® parameters of the µA 741 can be used from Wooley et al. [15] and the default and typical values from Tuinenga [16] and a detailed list of the same is given at Appendix B.

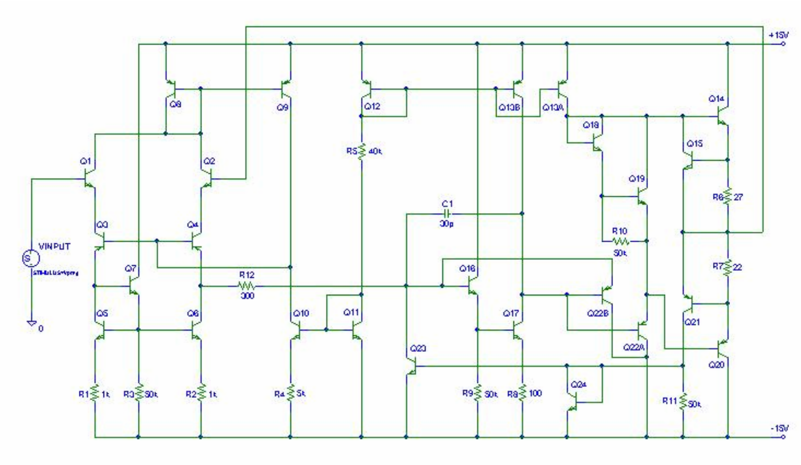


Figure 4.6: ***The schematic of the µA 741 operational amplifier circuit as a voltage follower.***

Applying the test PRN signal to the CUT under varying component values, signatures of fault-free and different faulty conditions can be generated in accordance with the flow diagram of Figure 4.5 using PSpice® and Monte-Carlo simulation technique. Non-faulty components are allowed to vary within their specified tolerance range in the Monte-Carlo analysis. Figure 4.7 shows the nominal output response and the extreme tolerance output response of the fault-free µA 741 Op-Amp connected as a voltage follower to the PRN test input. The data obtained from all these simulations were normalised and transferred to the MATLAB® environment for use in the simulation of the ANN models and generation of the fault signatures using the neural network toolbox [17]. A small multi-layer ANN with the input layer having one neuron using tan-sigmoid threshold function, a hidden layer of 3 neurons using tan-sigmoid threshold function and an output layer of one neuron using linear threshold function trained with BEP algorithm was used to model the fault-free as well as all the faulty classes. The tan-sigmoid function is used to map a neuron input from the interval (−∞, +∞) into the interval (−1, +1). The tan-sigmoid is a fully differentiable function, which makes it suitable for neurons being trained with back-propagation.

As stated in section 4.5.1 above, two models are constructed for the fault-free as well as for each type of fault: the nominal model and the model representing the extreme tolerance. Using the outputs of these models to the excitation by the PRN test signal, residuals are generated for fault-free as well as different types of fault. All these residuals for known fault conditions, known as fault signatures, are stored in a model bank for future reference. The output responses of the nominal ANN model and the extreme tolerance ANN model, and the nominal error output and the extreme tolerance error output for the fault-free µA 741 Op-Amp are also shown alongside in Figure 4.7.

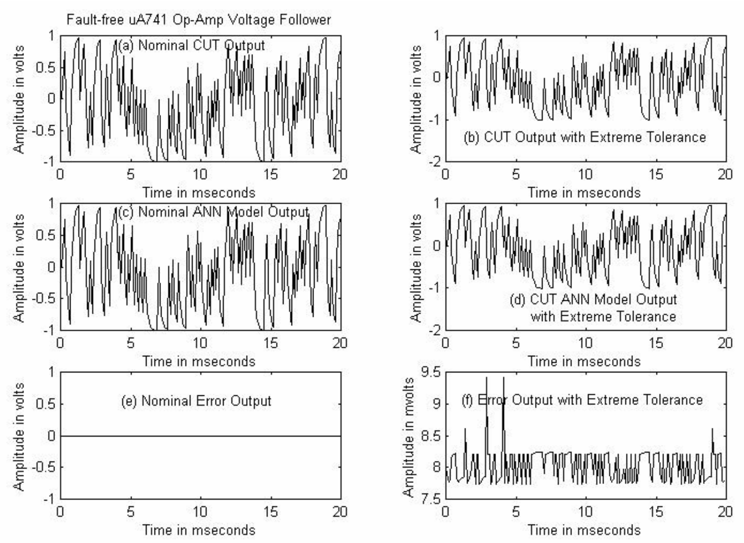
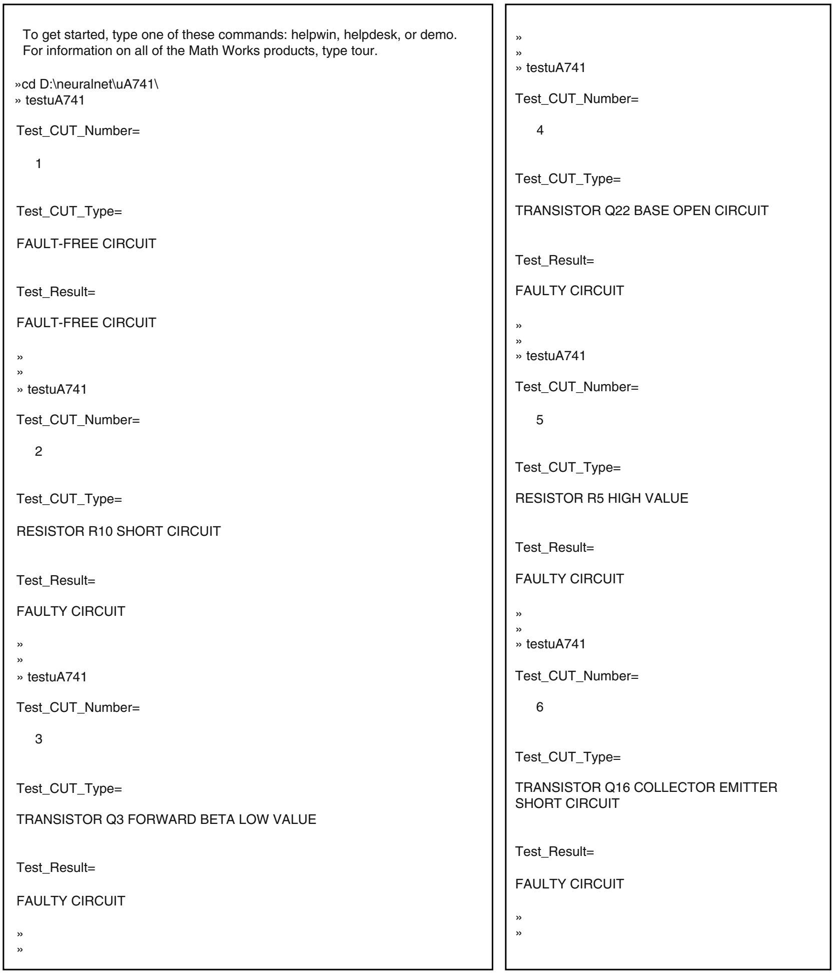


Figure 4.7: ***The nominal output response, the extreme tolerance output response, the nominal ANN Model response, the extreme tolerance ANN model response, the nominal error output and the extreme tolerance error output of the fault-free µA 741 Op-Amp voltage follower (Fig. 4.6).***

At the time of testing, the first step is to create an ANN model of the CUT. This is done by using the output obtained from the CUT as the target vector for the ANN. The output of the model is then compared to the output of the nominal fault-free ANN model by simulating both the models using the same PRN. This generates the residual (signature) for the circuit under test. The signature so obtained is now compared with the nominal fault-free signature stored in the model bank for the detection of faults, if any. Figure 4.8 shows the output displayed on the MATLAB® command window for the session results of the go/no-go testing algorithm written and stored in a MATLAB® m-file named testuA741.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the Math Works products, type tour.

»cd D:\neuralnet\uA741\
» testuA741

Test_CUT_Number=

    1


Test_CUT_Type=

FAULT-FREE CIRCUIT


Test_Result=

FAULT-FREE CIRCUIT

»
»
» testuA741

Test_CUT_Number=

    2


Test_CUT_Type=

RESISTOR R10 SHORT CIRCUIT


Test_Result=

FAULTY CIRCUIT

»
»
» testuA741

Test_CUT_Number=

    3


Test_CUT_Type=

TRANSISTOR Q3 FORWARD BETA LOW VALUE


Test_Result=

FAULTY CIRCUIT

»
»
```

```
»
»
» testuA741

Test_CUT_Number=

    4


Test_CUT_Type=

TRANSISTOR Q22 BASE OPEN CIRCUIT


Test_Result=

FAULTY CIRCUIT

»
»
» testuA741

Test_CUT_Number=

    5


Test_CUT_Type=

RESISTOR R5 HIGH VALUE


Test_Result=

FAULTY CIRCUIT

»
»
» testuA741

Test_CUT_Number=

    6


Test_CUT_Type=

TRANSISTOR Q16 COLLECTOR EMITTER
SHORT CIRCUIT


Test_Result=

FAULTY CIRCUIT

»
»
```

Figure 4.8: ***A session result of the go/no-go testing algorithm (testuA741.m) for the μA 741 Op-Amp.***

In this algorithm, the error output obtained from an observed (actual) model of the μA 741 is compared with the fault-free error signals of the μA 741 to detect the presence or absence of a fault. A circuit is defined to be faulty if the observed error output falls outside the band of nominal and extreme tolerance fault-free error signals of the μA 741 and fault-free if it remains within this band. As can be seen from this figure the error output obtained from an actual model of the fault-free μA 741 remains within the nominal and the extreme tolerance fault-free error signals indicating the circuit to be fault-free, while any error obtained from a faulty µA 741 falls outside this band indicating the circuit to be faulty. The faults induced here are chosen randomly and found to satisfy the algorithm to detect both fault-free and faulty circuits.

Once the go/no-go testing is complete and a fault is detected, the CUT is tested for classifying the fault using the fault identification and classification algorithm developed. In this algorithm, the error output obtained from an observed (actual) model of the µA 741 is compared with all possible fault signatures (error signals) of the µA 741 stored in the model bank to classify a fault. A fault is classified to be of a specific type if the observed error output falls within the band of nominal and extreme tolerance error outputs of a specific type of fault signature. It was observed that this simple algorithm could not diagnose many of the faults without introducing ambiguity.

To obtain a non-ambiguous, unique classification of the fault type, cross-correlation coefficients of the fault signatures with the nominal fault-free signature were evaluated and used. The cross-correlation coefficient, ρ_{xy}, is defined by

$$\rho_{xy} = \frac{Cov(x,y)}{\sigma_x \cdot \sigma_y} \tag{4.3}$$

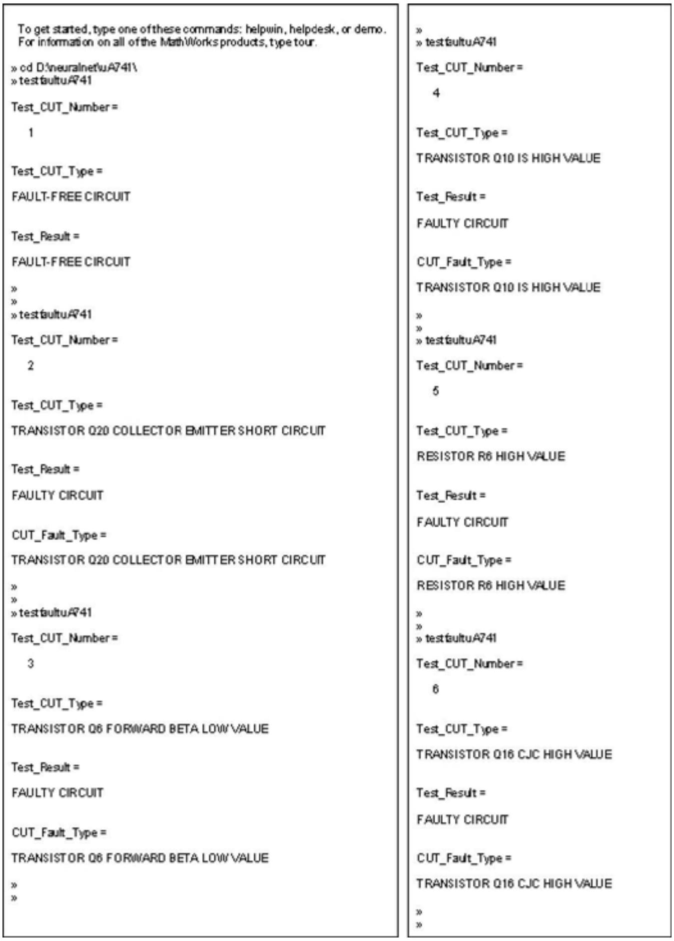
σ_x and σ_y being the standard deviations of the random variables x and y and are the positive square roots of their variances, respectively and $Cov(x,y)$ is the covariance between x and y. The covariance is given by the relationship

$$Cov(x,y) = R_{xy} - E[x] \cdot E[y] \tag{4.4}$$

where R_{xy} is the cross-correlation between x and y, and $E[x]$ and $E[y]$ are the expected values (or mean values) of the random variables x and y respectively.

Thus when a fault is detected using the go/no-go algorithm the cross-correlation coefficient of the observed fault signature with the nominal fault signature as well as those of all possible fault signatures stored in the model bank with that of the nominal fault-free signature were found out. Now a fault is classified to belong to a particular type if the cross-correlation coefficient of the observed fault signature with that of the nominal fault-free signature falls within the band of nominal and extreme tolerance values of the cross-correlation coefficients of a particular type of fault signature with that of the nominal fault-free signature. Figure 4.9 shows the output displayed on the MATLAB® command window for the session results obtained from the fault detection and diagnosis algorithm developed and stored in a MATLAB® m-file testfaultuA741.

Examples of the nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of a few faulty conditions of the µA 741 Op-Amp are given in Figures 4.10.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\uA741\
» testfaultuA741

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testfaultuA741

Test_CUT_Number =

    2


Test_CUT_Type =

TRANSISTOR Q20 COLLECTOR EMITTER SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

TRANSISTOR Q20 COLLECTOR EMITTER SHORT CIRCUIT

»
»
» testfaultuA741

Test_CUT_Number =

    3


Test_CUT_Type =

TRANSISTOR Q6 FORWARD BETA LOW VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

TRANSISTOR Q6 FORWARD BETA LOW VALUE

»
»
```

```
»
» testfaultuA741

Test_CUT_Number =

    4


Test_CUT_Type =

TRANSISTOR Q10 IS HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

TRANSISTOR Q10 IS HIGH VALUE

»
»
» testfaultuA741

Test_CUT_Number =

    5


Test_CUT_Type =

RESISTOR R6 HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R6 HIGH VALUE

»
»
» testfaultuA741

Test_CUT_Number =

    6


Test_CUT_Type =

TRANSISTOR Q16 CJC HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

TRANSISTOR Q16 CJC HIGH VALUE

»
»
```

Figure 4.9: ***A session result of the fault classification algorithm (testfaultuA741.m) for the μA 741 Op-Amp.***

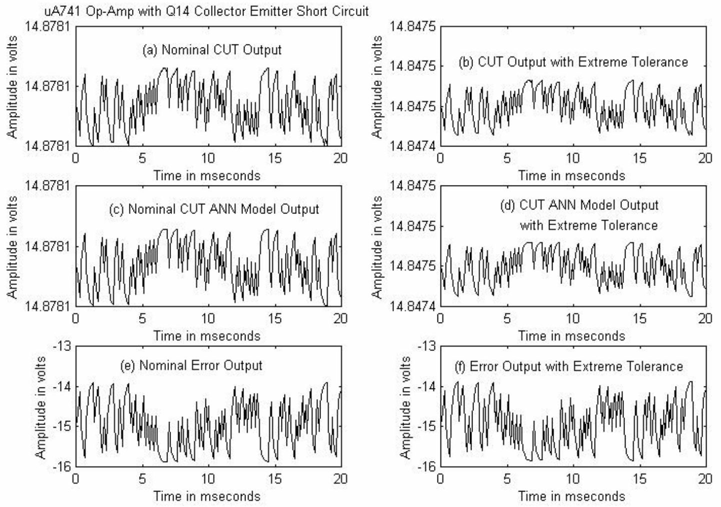


***Figure 4.10:** (i) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the μA 741 Op-Amp voltage follower (Fig. 4.6) faulted with Q14 collector emitter short circuit.*

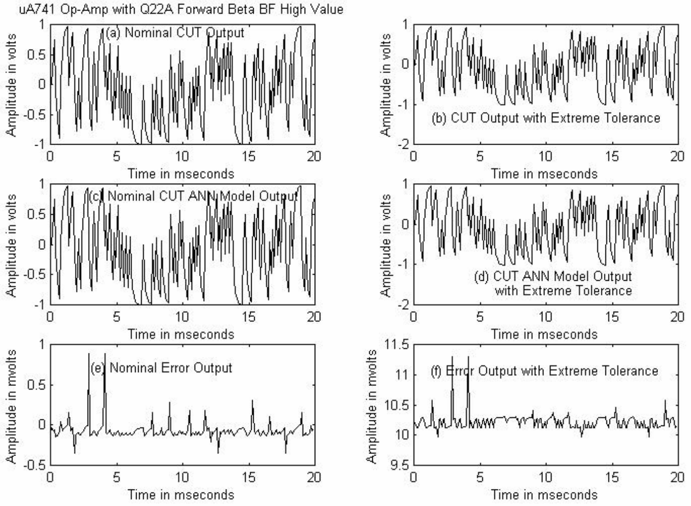


***Figure 4.10:** (ii) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the μA 741 Op-Amp voltage follower (Fig. 4.6) faulted with Q22A forward β BF high value.*

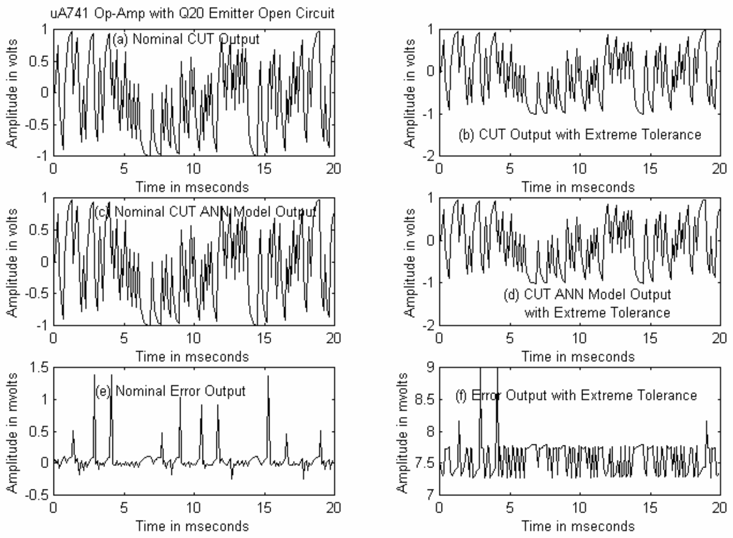


Figure 4.10: (iii) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the μA 741 Op-Amp voltage follower (Fig. 4.6) faulted with Q20 emitter open circuit.

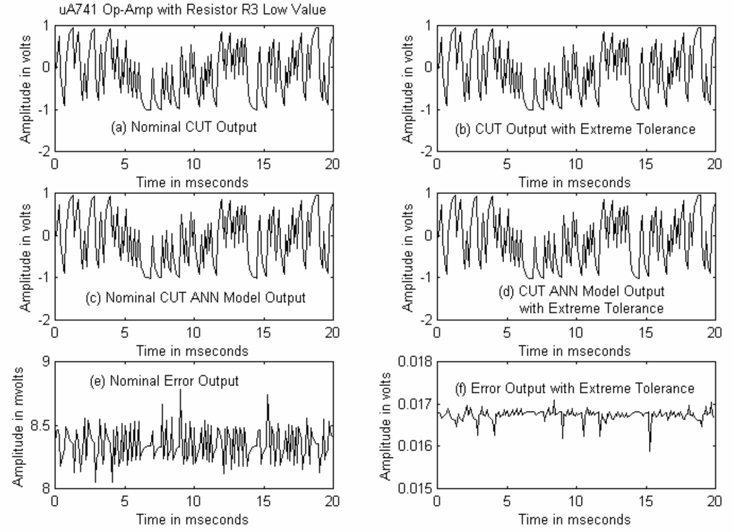


Figure 4.10: (iv) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the μA 741 Op-Amp voltage follower (Fig. 4.6) faulted with resistor R3 low value.

4.5.1.2 Example 2: The MOS Op-Amp

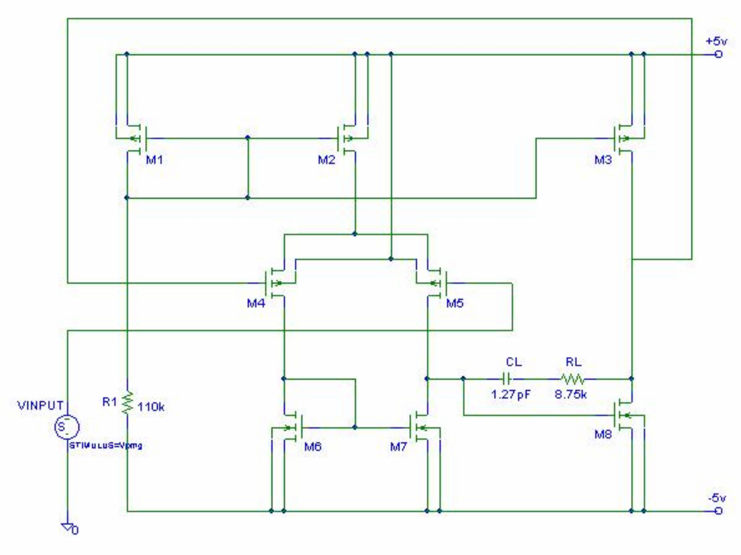


***Figure 4.11:** The schematic of the MOS operational amplifier circuit as a voltage follower.*

The schematic representation of a two-stage MOS Op-Amp used in the voltage follower configuration is shown in Figure 4.11. The Op-Amp has been designed using 1.2µm CMOS process parameters form MITEL Semiconductor. The nominal values of the circuit components are:

R1 = 110kΩ, RL = 8.75kΩ, CL = 1.27pF.

The components are expected to vary in their parameter values due to manufacturing tolerances and variations in voltage, temperature, etc. The variations in component values will cause variation in the output response of the circuit to an input test stimulus. In turn, variations in the output response are considered to determine whether a fault has occurred or not. In our studies, all the parameters were assigned ±5% tolerance.

For complete diagnosis, a large number of fault classes are required to be defined depending on the number of components and the level of diagnostics. The MOS Op-Amp voltage follower circuit was simulated using PSpice® and Monte-Carlo analysis for fault-free and different faulty conditions with PRN as its input stimulus as per the flow diagram of Figure 4.5. Non-faulty components were allowed to vary within their specified tolerance range, which established the signatures for fault-free CUT. As defined in section 4.5.1, Figure 4.12 shows the nominal output response and the extreme tolerance output response of the fault-free MOS Op-Amp.

The default and typical values of the PSpice® parameters of the CMOS Op-Amp can be used from Tuinenga [16] and those for the nominal values from the IEEE mixed-signal benchmark circuit homepage [18] at: http://faculty.washington.edu/manisoma/madtest/benchmarks/OpAmp.htm. A detailed list of the PSpice® parameters of the CMOS Op-Amp is given in Appendix C.

As stated before the data obtained from these simulations were normalised before being transferred and used in the simulation of the ANN model using MATLAB® neural network toolbox. On completion of this task, the fault signatures for various faults indicated above were generated and stored in a fault signature bank. The output responses of the nominal ANN model and the extreme tolerance ANN model, and the nominal error output and the extreme tolerance error output for the fault-free MOS Op-Amp are also shown alongside in Figure 4.12.

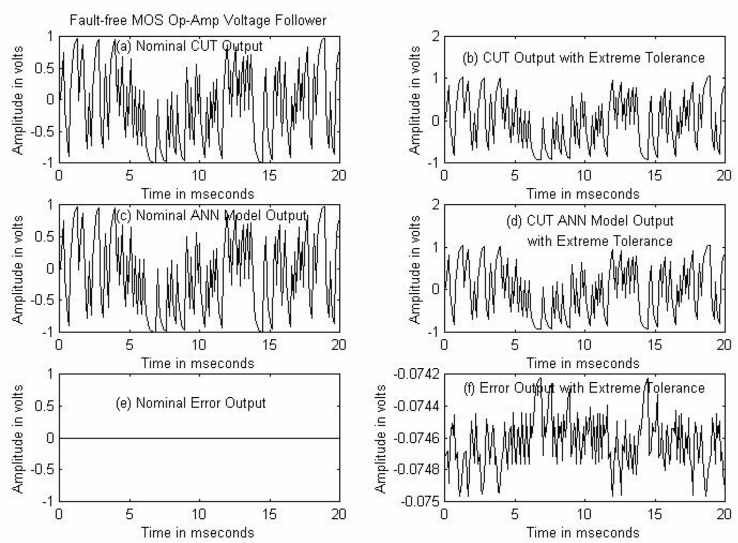
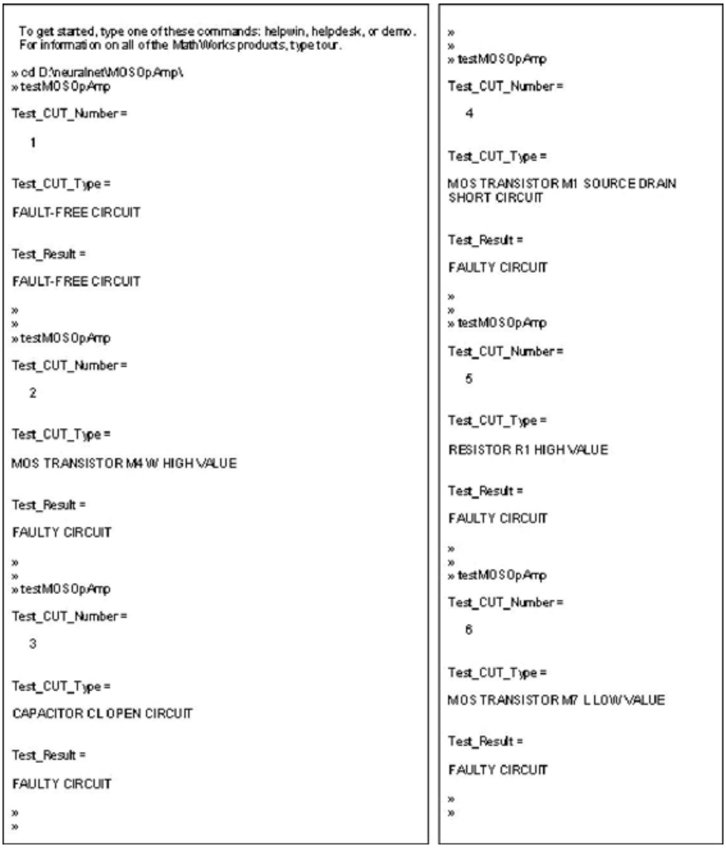


Figure 4.12: ***The nominal output response, the extreme tolerance output response, the nominal ANN Model response, the extreme tolerance ANN model response, the nominal error output and the extreme tolerance error output of the fault-free MOS Op-Amp voltage follower (Fig. 4.11).***

Next step is to test the diagnostic system for detection and classification of faults in the MOS Op-Amp. For these, faults were induced in the MOS Op-Amp randomly. It was observed that all the faults were detectable. Figure 4.13 shows the output represented in the MATLAB® command window for the session results of the go/no-go testing algorithm developed and stored in the MATLAB® m-file testMOSOpAmp. As can be seen from this figure the error output obtained from an actual model of the fault-free CUT, when compared with the fault-free signatures (error signals) indicate the circuit to be fault-free, while any error obtained from an actual model of a faulty CUT indicated occurrence of fault.

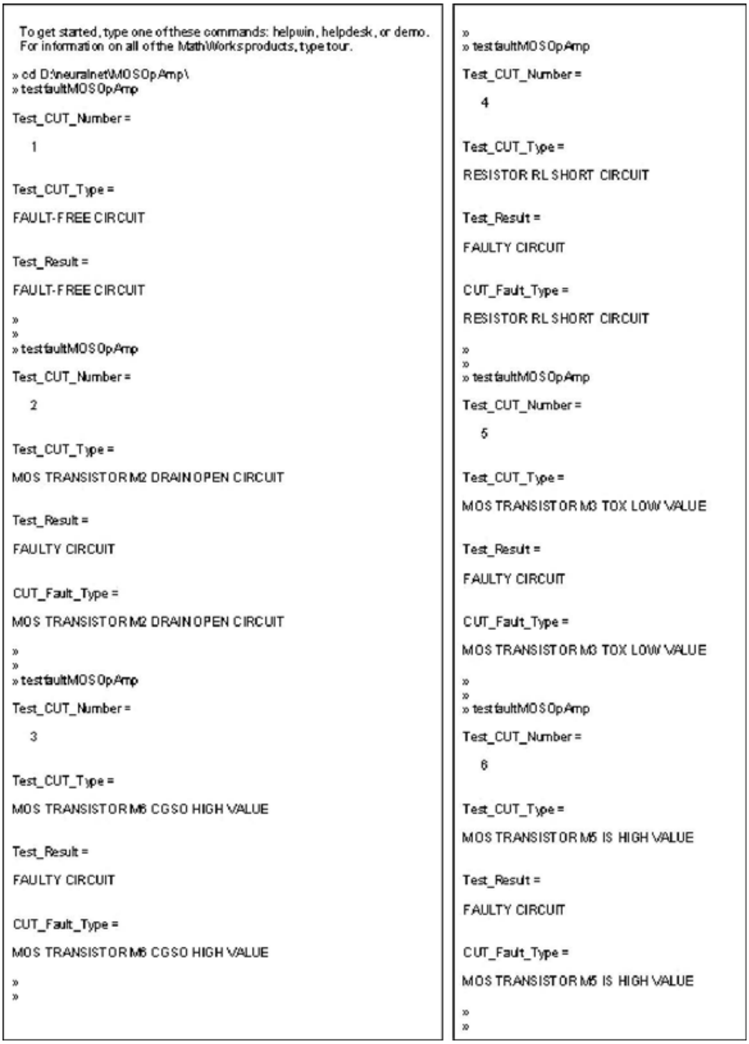


```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\MOSOpAmp\
» testMOSOpAmp

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testMOSOpAmp

Test_CUT_Number =

    2


Test_CUT_Type =

MOS TRANSISTOR M4 W HIGH VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testMOSOpAmp

Test_CUT_Number =

    3


Test_CUT_Type =

CAPACITOR CL OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
```

```
»
»
» testMOSOpAmp

Test_CUT_Number =

    4


Test_CUT_Type =

MOS TRANSISTOR M1  SOURCE DRAIN
SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testMOSOpAmp

Test_CUT_Number =

    5


Test_CUT_Type =

RESISTOR R1 HIGH VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testMOSOpAmp

Test_CUT_Number =

    6


Test_CUT_Type =

MOS TRANSISTOR M7 L LOW VALUE


Test_Result =

FAULTY CIRCUIT

»
»
```

Figure 4.13: A session result of the go/no-go testing algorithm (testMOSOpAmp.m) for the MOS Op-Amp.

If a fault is detected, the error signature is further tested for classifying the fault using the fault classification algorithm. The fault classification algorithm, as discussed before, uses the cross-correlation coefficients between the observed fault signature and the nominal fault-free signature as well as those of all possible fault signatures stored in the model bank with that of the nominal fault-free signature. This is done to avoid ambiguity in the fault classification as some of the residuals may fall within the same range of values. Some outputs of the MATLAB® command window session results obtained from this algorithm stored in the MATLAB® m-file testfaultMOSOpAmp are shown in Figure 4.14.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\MOSOpAmp\
» testfaultMOSOpAmp

Test_CUT_Number =

    1

Test_CUT_Type =

FAULT-FREE CIRCUIT

Test_Result =

FAULT-FREE CIRCUIT

»
»
» testfaultMOSOpAmp

Test_CUT_Number =

    2

Test_CUT_Type =

MOS TRANSISTOR M2 DRAIN OPEN CIRCUIT

Test_Result =

FAULTY CIRCUIT

CUT_Fault_Type =

MOS TRANSISTOR M2 DRAIN OPEN CIRCUIT

»
»
» testfaultMOSOpAmp

Test_CUT_Number =

    3

Test_CUT_Type =

MOS TRANSISTOR M6 CGSO HIGH VALUE

Test_Result =

FAULTY CIRCUIT

CUT_Fault_Type =

MOS TRANSISTOR M6 CGSO HIGH VALUE

»
»
```

```
»
» testfaultMOSOpAmp

Test_CUT_Number =

    4

Test_CUT_Type =

RESISTOR RL SHORT CIRCUIT

Test_Result =

FAULTY CIRCUIT

CUT_Fault_Type =

RESISTOR RL SHORT CIRCUIT

»
»
» testfaultMOSOpAmp

Test_CUT_Number =

    5

Test_CUT_Type =

MOS TRANSISTOR M3 TOX LOW VALUE

Test_Result =

FAULTY CIRCUIT

CUT_Fault_Type =

MOS TRANSISTOR M3 TOX LOW VALUE

»
»
» testfaultMOSOpAmp

Test_CUT_Number =

    6

Test_CUT_Type =

MOS TRANSISTOR M5 IS HIGH VALUE

Test_Result =

FAULTY CIRCUIT

CUT_Fault_Type =

MOS TRANSISTOR M5 IS HIGH VALUE

»
»
```

Figure 4.14: A session result of the fault classification algorithm (testfaultMOSOpAmp.m) for the MOS Op-Amp.

The nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of a few faulty conditions of the MOS Op-Amp are depicted in Figures 4.15.

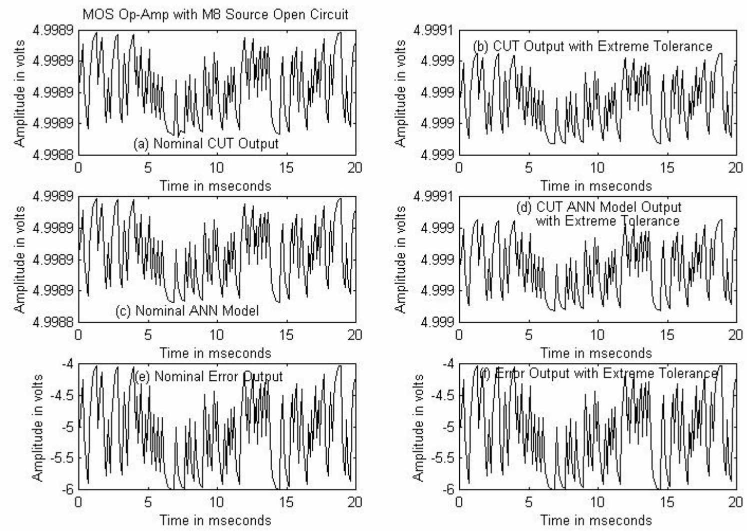


***Figure 4.15:** (i) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the MOS Op-Amp voltage follower (Fig. 4.9) faulted with M8 source open circuit.*

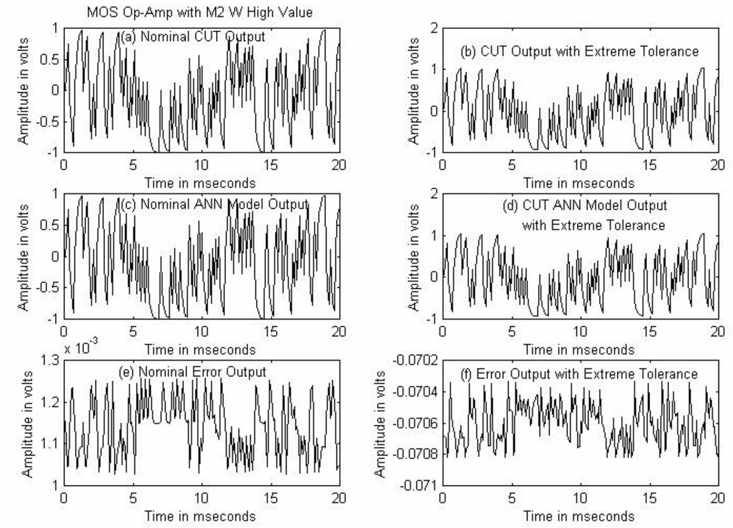


Figure 4.15: ***(ii) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the MOS Op-Amp voltage follower (Fig. 4.9) faulted with M2 channel width W high value.***

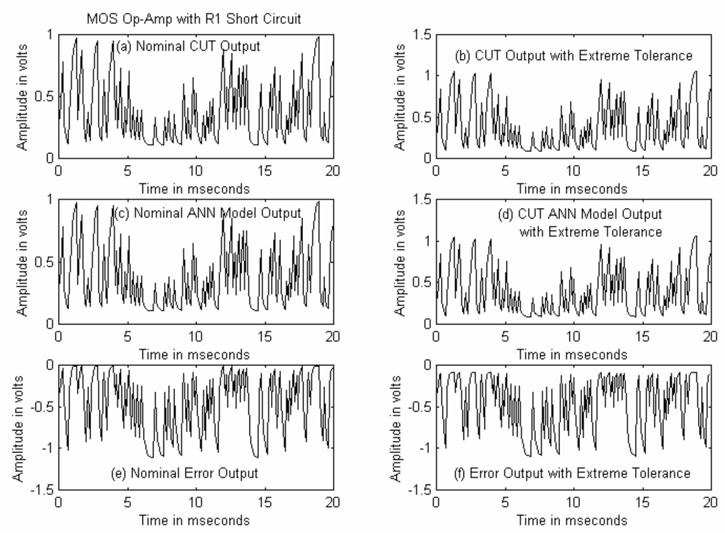


Figure 4.15: ***(iii) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the MOS Op-Amp voltage follower (Fig. 4.9) faulted with Resistor R1 short circuit.***

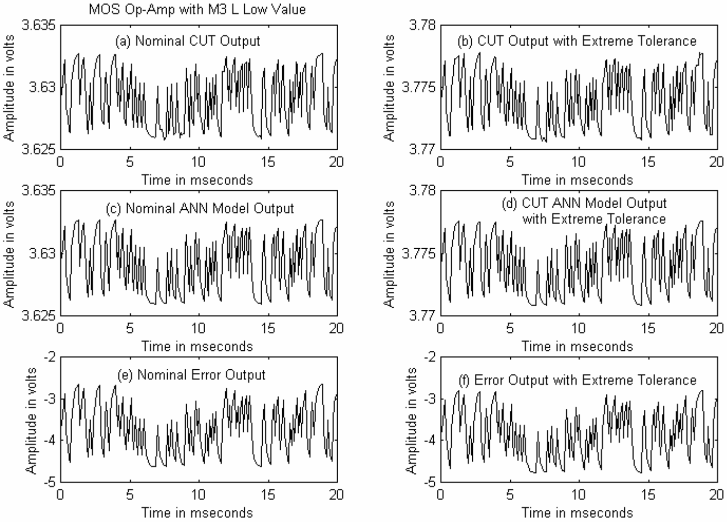


Figure 4.15: ***(iv) The nominal output response, the extreme tolerance output response, the nominal ANN Model output, the extreme tolerance ANN model output, the nominal error output and the extreme tolerance error output of the MOS Op-Amp voltage follower (Fig. 4.9) faulted with M3 channel length L low value.***

4.5.2 Fault diagnosis in embedded analog IC

As stated before, fault diagnosis in analog integrated circuits (ICs) is complicated owing to the continuous nature of their input and output signals, tolerance and non-linearity of the circuit elements and the difficulty in modelling analog behaviour due to poor availability of fault models of analog devices and components. Test and diagnosis mechanisms require sufficient controllability and observability of the fault model. A direct application of the digital *stuck-at* fault model proves to be inadequate for the analog-circuit fault simulation and test vector generation. Moreover, only limited numbers of test stimulus and output points are available in analog ICs. In general, it is not possible to apply test signal or observe a response at an arbitrary point in an analog IC. This further decreases the fault coverage. An ideal strategy would allow all faults to be detectable through the primary outputs only. Part of the solution lies with proper selection of the test signal.

This section describes testing and fault diagnosis of embedded analog ICs based on model based observer scheme. The steps involved in the procedure of observer-based fault diagnosis of embedded analog integrated circuits (based on the general procedure of observer-based fault diagnosis described in Section 4.5) are the following:

- defining the faults of interest,
- simulation of the circuit to generate response signatures for the defined types of fault,
- generation of residuals,
- detection of fault, and
- classification of the fault.

The key to the observer-based fault diagnosis procedure is the generation and evaluation of a set of residuals that permits not only the detection but also discrimination between different faults. In general, a bank of observers or an observation scheme, where each observer/observation is made sensitive to a particular fault or a particular group of faults, whilst maintaining insensitivity to unmodelled disturbances, noise, modelling uncertainties and parameter variations within permissible limits, could achieve this goal.

Refer to section 3.5; the test signal used is the pseudo-random noise (PRN) signal. Such a signal maintains the same range of amplitude throughout the frequency spectrum of interest. The output power spectral density of the circuit under test, P_{out}, to this PRN signal, having an input power spectral density, η, is given by,

$$P_{out} = \eta \left| H(j\omega) \right|^2, \tag{4.5}$$

where $\left| H(j\omega) \right|$ is the frequency response of the circuit under test (CUT). In practice, Equation (4.4) will hold if the test PRN signal has a constant spectrum over a bandwidth sufficiently larger than that of the CUT. The advantage of using the PRN as the test signal is that hopefully it will excite all the modes of the circuit and a complete frequency response of the circuit would be observed at output.

The analog circuit under test is simulated using PSpice® and Monte-Carlo techniques for fault-free and faulty conditions using PRN as the input test stimulus as per the flow diagram of Figure 4.5. Non-faulty components are allowed to vary within their specified tolerance range to establish the signature for fault-free CUT. A multi-layer ANN model trained with back-error propagation algorithm is used as before to model the fault-free as well as the faulty circuits, using the neural network toolbox in MATLAB® [17]. As discussed in section 4.5.1, two models are constructed for the fault-free as well as for each type of fault. One is the nominal model representing the behaviour of the circuit with all parameters held at their nominal values; the second model represent the behaviour of the circuit for maximum deviation in its output response during the Monte-Carlo simulation of the CUT taking into account extreme variations of the components within their tolerance range. For generation of residuals, these models are excited by the PRN test signal and the output compared to the output of the nominal fault-free model excited by the same PRN. The difference constitutes the error signal, which is the residual for the particular type of fault. All these residuals for known fault conditions, known as fault signatures, are stored in a model bank for future reference.

At the time of testing, the first step is to create an ANN model of the CUT. This is done by using the output obtained from the CUT as the target vector for the ANN. The output of the model is then compared to the output of the nominal fault-free ANN model by exciting both the models using the same PRN. This generates the residual (signature) for the circuit under test. The signature so obtained is now compared with the nominal fault-free signature stored in the model bank for the detection of fault, if any. When there is no fault in the CUT, the two models (the nominal and the actual) will present a very close behaviour and will output little or no residual. Under faulty condition, the CUT will behave differently and so will the ANN model of the CUT. The system will now output an error signal (residual) showing how much the actual CUT is different from the fault-free CUT. Any fault in the CUT, be it catastrophic or parametric, will change the signature. Signatures of known fault conditions of the CUT are first simulated and stored in a model bank for future reference. Making provisions for common fault signatures develops to a fast, predetermined fault diagnostics.

For testing validity of the observer-based methodology of fault diagnosis in embedded integrated circuits, two examples were selected. One is a single amplifier band-pass filter [19] using the µA 741 Op-Amp. This network is one of the most commonly used filter network and popularly known as the Sallen and Key (S-K) filter. The other example chosen is the simple three amplifier state variable filter from the IEEE Analog and Mixed-Signal Benchmark Circuits [14].

4.5.2.1 Example 1: The single amplifier bandpass filter

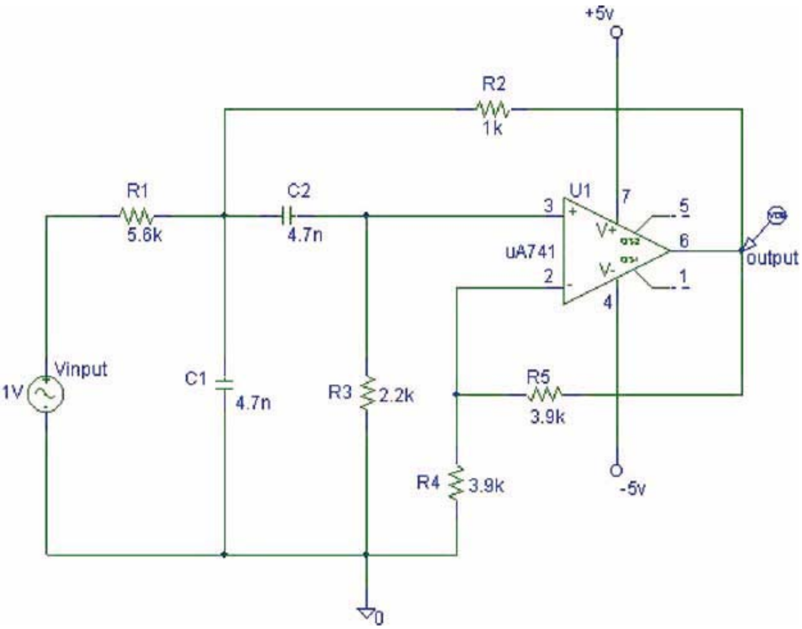


Figure 4.16: The single amplifier band-pass filter circuit.

The schematic of the single amplifier band-pass test circuit is shown in Figure 4.16. The nominal values of the circuit components are as given below:

R1 = 5.6kΩ, R2 = 1kΩ, R3 = 2.2kΩ, R4 = R5 = 3.9kΩ, C1 = C2 = 4.7nF. All the components were assigned ±5% tolerance.

The circuit of Figure 4.16 is known as the Sallen and Key band-pass filter. The voltage transfer function of the circuit is given by

$$H(s) = \frac{V_o(s)}{V_{in}(s)}$$

$$= \frac{\dfrac{ks}{R_1C_1}}{s^2 + \left(\dfrac{1}{R_1C_1} + \dfrac{1}{R_3C_2} + \dfrac{1}{R_3C_1} + \dfrac{1-k}{R_2C_1}\right)s + \dfrac{R_1+R_2}{R_1R_2R_3C_1C_2}}, \tag{4.6}$$

where $k = 1 + \dfrac{R_5}{R_4}$.

Comparing equation (4.5) with the second order band-pass filter transfer function

$$H(s) = K\frac{s}{s^2 + (\omega_p / Q_p)s + \omega_p^2}, \tag{4.7}$$

where ωp, Qp and K are the pole frequency, pole selectivity and gain constant of the filter and expressed as

$$\omega_p = \sqrt{\frac{R_1+R_2}{R_1R_2R_3C_1C_2}} \tag{4.8a}$$

$$Q_p = \frac{\sqrt{\dfrac{R_1+R_2}{R_1R_2R_3C_1C_2}}}{\dfrac{1}{R_1C_1} + \dfrac{1}{R_3C_1} + \dfrac{1}{R_3C_2} + \dfrac{1-k}{R_2C_1}} \tag{4.8b}$$

$$K = \frac{k}{R_1C_1} \tag{4.8c}$$

With the nominal values of the components the centre frequency of the band-pass filter is $f_p \cong 25kHz$. The frequency response of the band-pass filter circuit obtained by PSpice simulation is shown in Figure 4.17. It is observed that the centre frequency of this circuit is 23.26kHz. The discrepancy could be due to the finite gain bandwidth product of the Op-Amp.

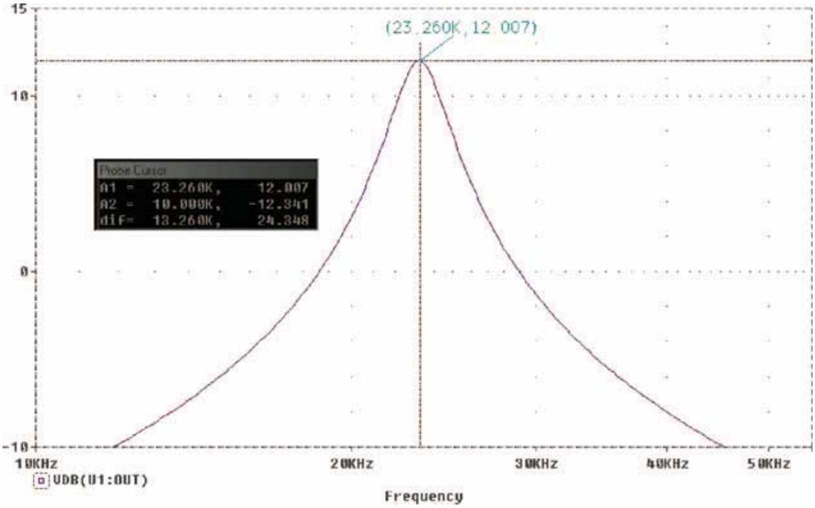


Figure 4.17: The frequency response graph of the single amplifier band-pass filter circuit (Fig. 4.16).

As stated previously in section 4.5.2, fault signatures were generated by simulating the circuit in PSpice using Monte-Carlo analysis to the application of the PRN test signal under varying fault conditions. The tolerance assigned for the Monte-Carlo analysis was sufficient to bring the component value to the boundary of the nominal value plus tolerance. For example, a nominal component may have a value of 100Ω with 5% tolerance. This means the valid fault-free operating range is from 95Ω to 105Ω. For fault on the higher side, simulations for values of the resistor from 106Ω onwards were performed with reasonable increments. Similarly, the value of the faulty component was defined on the lower side to produce a faulty range of resistor values below 95Ω.

Data obtained from the PSpice® and Monte-Carlo simulations are normalised and transferred to the MATLAB® environment for further processing. A multi-layer ANN trained with BEP algorithm was used to model the fault-free as well as each of the faulty circuits using the neural network toolbox. The ANN consists of an input layer having 1 input neuron with the tan sigmoid threshold function, 2 hidden layers having 15 neurons each with the tan sigmoid threshold function, and an output layer having one output neuron with a linear threshold function. Instead of the simple three-layer ANN model for the single amplifier band-pass filter, a more-deep network is used to reduce training time and satisfy the desired error goal. As stated previously, two models were constructed for the fault-free as well as for each type of fault: the nominal and the extreme tolerance model. For generation of residuals, these models were excited by the PRN test signal and the output compared to the output of the nominal fault-free model excited by the same PRN. The difference constitutes the error signal, which is the residual for the particular type of fault. All these residuals for known fault conditions, known as fault signatures, were stored in a model bank for future reference.

The nominal output response, the extreme tolerance output response, the nominal ANN model response, the extreme tolerance ANN model response, the nominal error output, and the extreme tolerance error output for the fault-free band-pass filter circuit are shown in Figure 4.18.

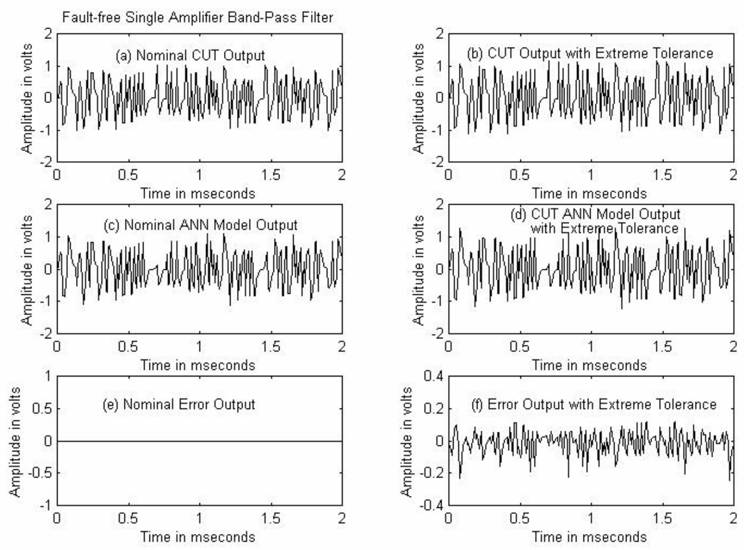
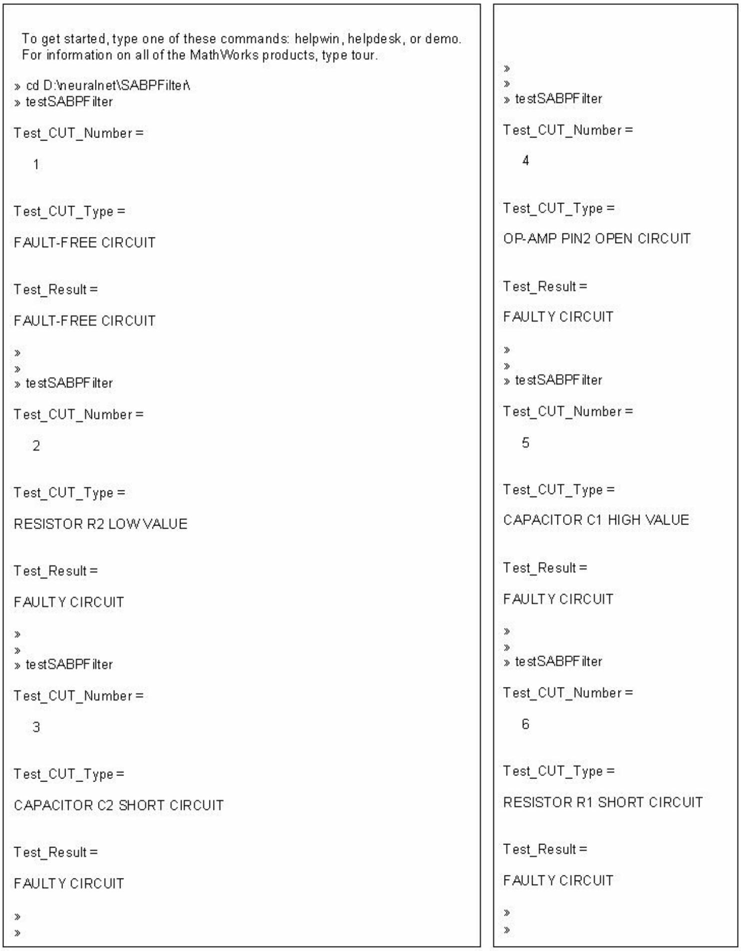


Figure 4.18: The nominal output response, the extreme tolerance output response, the nominal ANN model response, the extreme tolerance ANN model response, the nominal error output, and the extreme tolerance error output of the fault-free single amplifier band-pass filter (Fig. 4.16).

At the time of testing, the output obtained from the CUT is used as the target vector for the training of the ANN model. The output of this model is then compared to the output of the nominal fault-free ANN model by simulating both the models using the same PRN. This generates the residual (signature) for the CUT. The signature so obtained is now compared with the nominal fault-free signature stored in the model bank for the detection of faults, if any. Figure 4.19 gives session results of the go/no-go testing algorithm written and stored in a MATLAB® m-file named testSABPFilter.

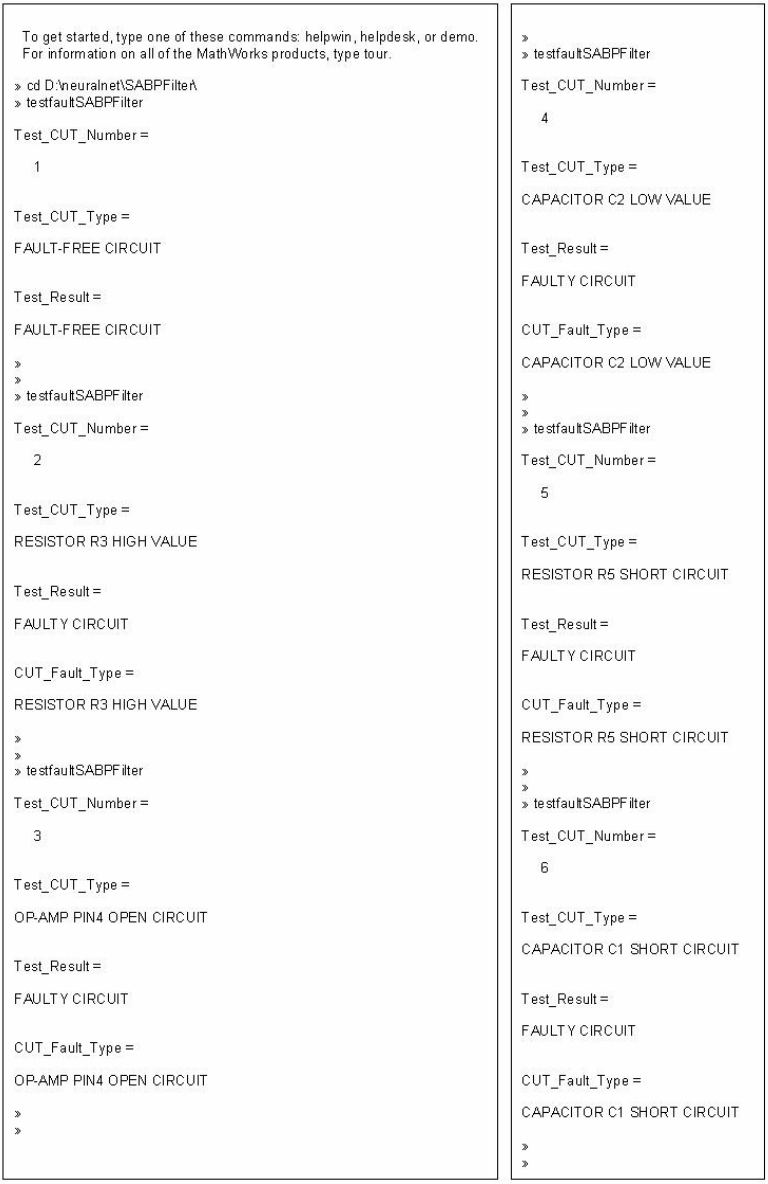


```
  To get started, type one of these commands: helpwin, helpdesk, or demo.
  For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SABPFilter\
» testSABPFilter

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    2


Test_CUT_Type =

RESISTOR R2 LOW VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    3


Test_CUT_Type =

CAPACITOR C2 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
```

```
»
»
» testSABPFilter

Test_CUT_Number =

    4


Test_CUT_Type =

OP-AMP PIN2 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    5


Test_CUT_Type =

CAPACITOR C1 HIGH VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    6


Test_CUT_Type =

RESISTOR R1 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
```

***Figure 4.19:** A session result of the go/no-go testing algorithm (testSABPFilter.m) for the single amplifier band-pass filter.*

Once a fault is detected in the go/no-go testing described above, the CUT is tested for classification of the type of fault by comparing the signature with the fault signatures stored in the model bank. A fault is classified to be of a specific type if the observed error output falls within the band of nominal and extreme tolerance error outputs of a specific type of fault signature.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SABPFilter\
» testfaultSABPFilter

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testfaultSABPFilter

Test_CUT_Number =

    2


Test_CUT_Type =

RESISTOR R3 HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R3 HIGH VALUE

»
»
» testfaultSABPFilter

Test_CUT_Number =

    3


Test_CUT_Type =

OP-AMP PIN4 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

OP-AMP PIN4 OPEN CIRCUIT

»
»
```

```
»
» testfaultSABPFilter

Test_CUT_Number =

    4


Test_CUT_Type =

CAPACITOR C2 LOW VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C2 LOW VALUE

»
»
» testfaultSABPFilter

Test_CUT_Number =

    5


Test_CUT_Type =

RESISTOR R5 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R5 SHORT CIRCUIT

»
»
» testfaultSABPFilter

Test_CUT_Number =

    6


Test_CUT_Type =

CAPACITOR C1 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C1 SHORT CIRCUIT

»
»
```

Figure 4.20:* *A session result of the fault classification algorithm (testfaultSABPFilter.m) for the single amplifier band-pass filter.

It was observed that this simple algorithm could not diagnose many of the faults properly and gives rise to ambiguity. To obtain a non-ambiguous, unique classification of the fault type, cross-correlation coefficients of the fault signatures with the nominal fault-free signature were evaluated and used. Thus when a fault is detected using the go/no-go algorithm the cross-correlation coefficient between the observed fault signature and the nominal fault-free signature as well as those of all possible fault signatures stored in the model bank were found out. Now a fault is classified to belong to a particular type if the cross-correlation coefficient of the observed fault signature with that of the nominal fault-free signature falls within the band of nominal and extreme tolerance values of the cross-correlation coefficients of a particular type of fault signature with that of the nominal fault-free signature. Some of the session results obtained from the fault detection and diagnosis algorithm developed and stored in a MATLAB® m-file named testfaultSABPFilter, is recorded in Figure 4.20.

Examples of the nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of a few faulty conditions of the single amplifier band-pass filter circuit using µA 741 Op-Amp are given in Figures 4.21.

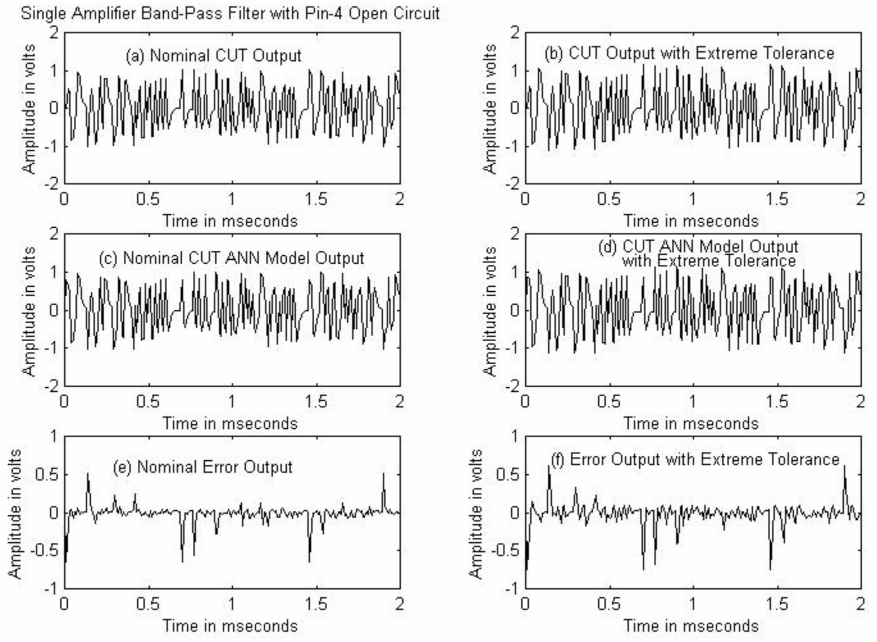


Figure 4.21: ***(i) The nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of the single amplifier band-pass filter (Fig. 4.16) with Op-Amp pin4 open circuit.***

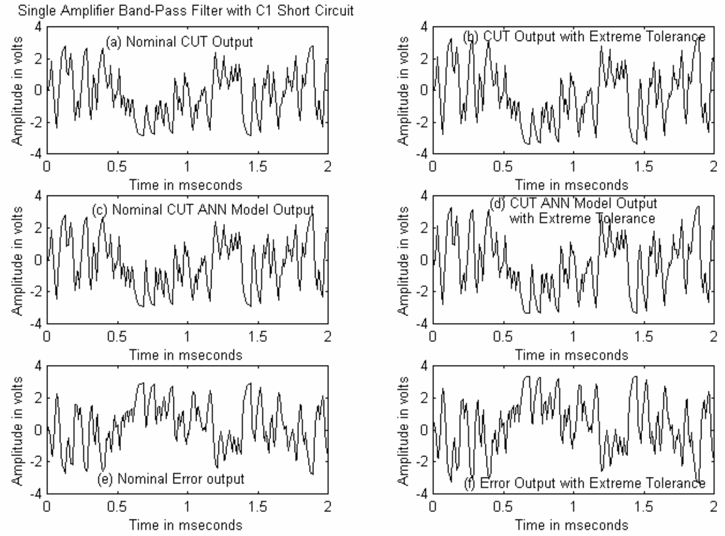


Figure 4.21: ***(ii) The nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of the single amplifier band-pass filter (Fig. 4.16) with capacitor C1 short circuit.***

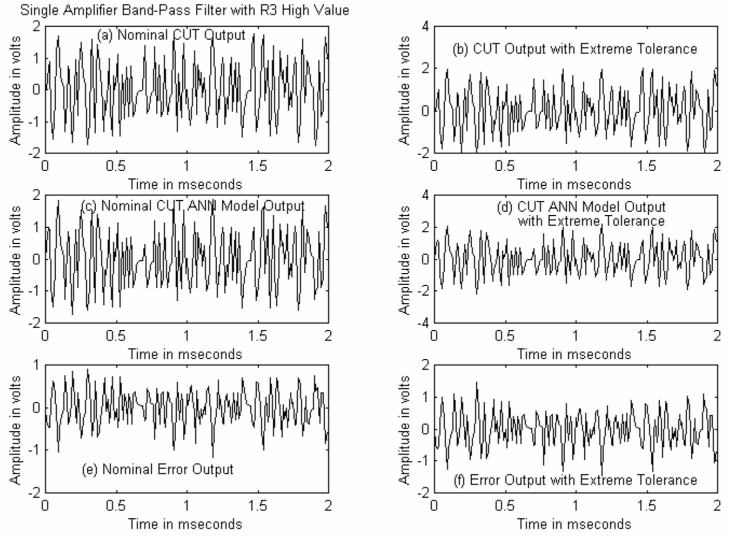


Figure 4.21: ***(iii) The nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of the single amplifier band-pass filter (Fig. 4.16) with Resistor R3 high value.***

4.5.2.2 Example 2: The three amplifier state variable filter

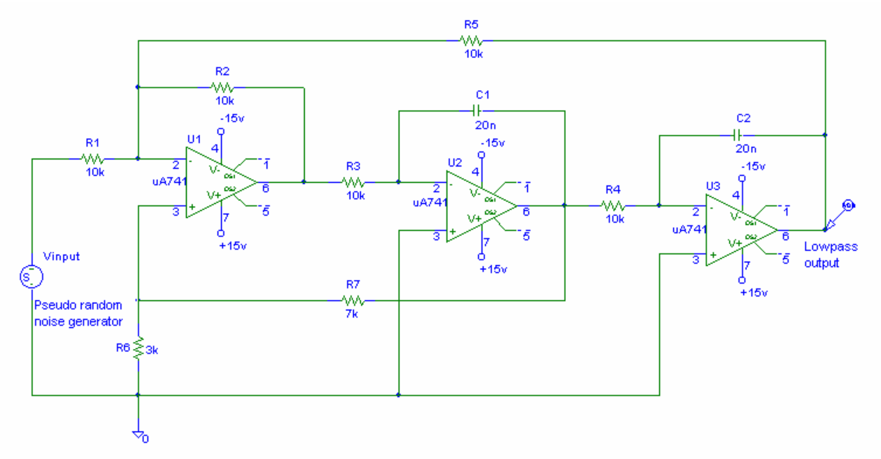


Figure 4.22: The state variable filter circuit.

A three amplifier state variable filter is shown in Figure 4.22. The circuit uses two Op-Amp based integrators and a summing amplifier to provide the second order low-pass, high-pass and band-pass responses. The nominal values of the circuit components are:

R1 = R2 = R3 = R4 = R5 = 10kΩ, R6 = 3kΩ, R7 = 7kΩ, C1 = C2 = 20nF. All the parameters were assigned ±5% tolerance.

The voltage transfer function of the second-order state-variable filter (Fig. 4.22), considering its low-pass output (LPO) is given by

$$\frac{V_{LPO}}{V_{input}} = -\frac{R_5}{R_1}\left[\frac{\dfrac{R_2/R_5}{R_3C_1R_4C_2}}{s^2 + \dfrac{(1+R_2/R_5+R_2/R_1)}{(1+R_7/R_6)R_3C_1}s + \dfrac{R_2/R_5}{R_3C_1R_4C_2}}\right] \tag{4.9}$$

Comparing Equation (4.9) with the second order low-pass filter transfer function, we get the following relations for K, ω$_0$ and Q.

$$K = R_5/R_1 \tag{4.10a}$$

$$\omega_0 = \sqrt{\frac{(R_2/R_5)}{(R_3C_1R_4C_2)}} \tag{4.10b}$$

$$Q = \frac{\sqrt{\{(R_2/R_5)(R_3C_1/R_4C_2)\}}(1+R_7/R_6)}{(1+R_2/R_5+R_2/R_1)} \tag{4.10c}$$

where K, ω₀ and Q represent the gain constant, the pole frequency and the pole selectivity of the filter respectively. The state variable filter is usually implemented with $R_1 = R_2 = R_5$, $R_3 = R_4 = R$, and $C_1 = C_2 = C$, so the expressions for K, ω_0, and Q simplify to

$$\mathrm{K} = 1,\ \omega_0 = 1/RC\text{, and } Q = \frac{1}{3}(1+R_7/R_6) \tag{4.11}$$

Therefore, for the LPO of Figure 4.22 with nominal values of the components, we have, K = 1.0, Q = 1.11, and $f_0 \cong 796$Hz. The frequency response of the filter obtained from the PSpice simulation is shown in Figure 4.23, which indicates the 3dB cut-off frequency, f_0, to be about 829Hz. This discrepancy could be due to the finite gain-bandwidth product of the Op-Amps.

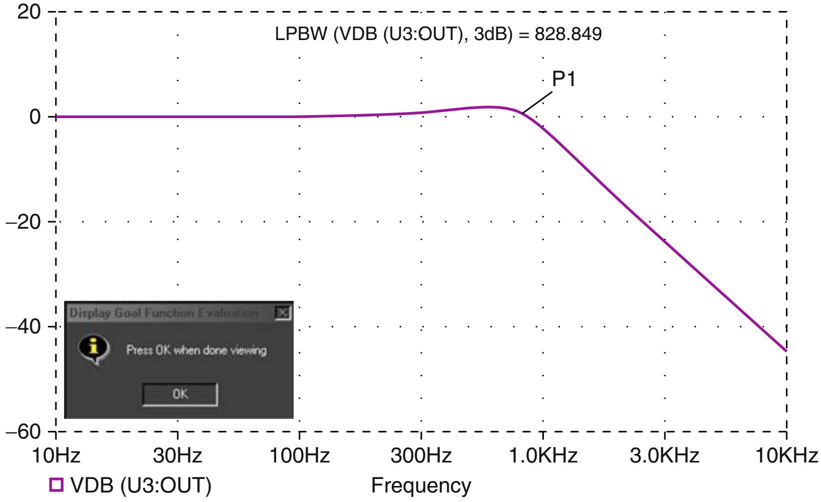


Figure 4.23: The frequency response graph of the state variable filter circuit (Fig. 4.22).

The circuits were simulated using PSpice and Monte-Carlo techniques for fault-free as well as faulty conditions using PRN as the input test stimulus. As stated before, non-faulty components were allowed to vary within their specified tolerance range to establish the signature for fault-free circuit. The nominal output response and the extreme tolerance output response of the fault-free circuit are shown in Figure 4.24. Catastrophic faults like short and open circuits and incidence of parametric faults introducing large variations in the circuit component values were introduced and simulated. As explained in the previous example the Monte-Carlo simulated data using PSpice were normalised and transferred to the MATLAB environment for training the ANN models. Signatures for different type of faults were obtained and stored in a model bank for comparison during actual testing of the circuit. The nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of the fault-free state variable filter are shown in Figure 4.24.

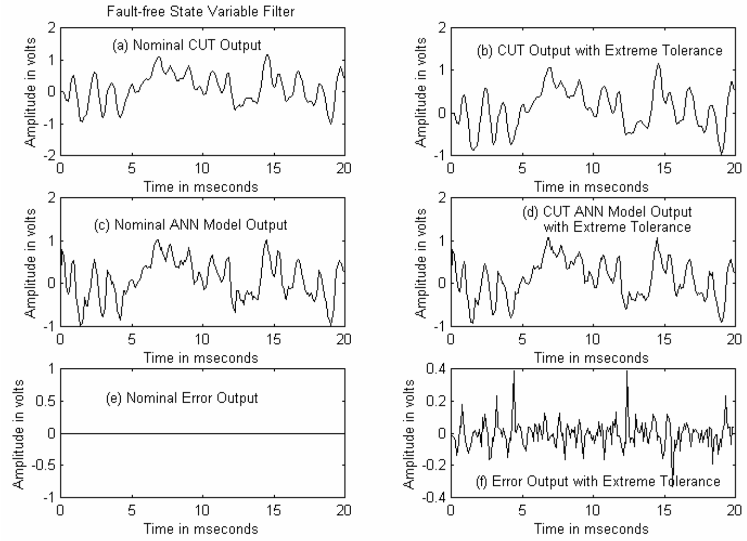
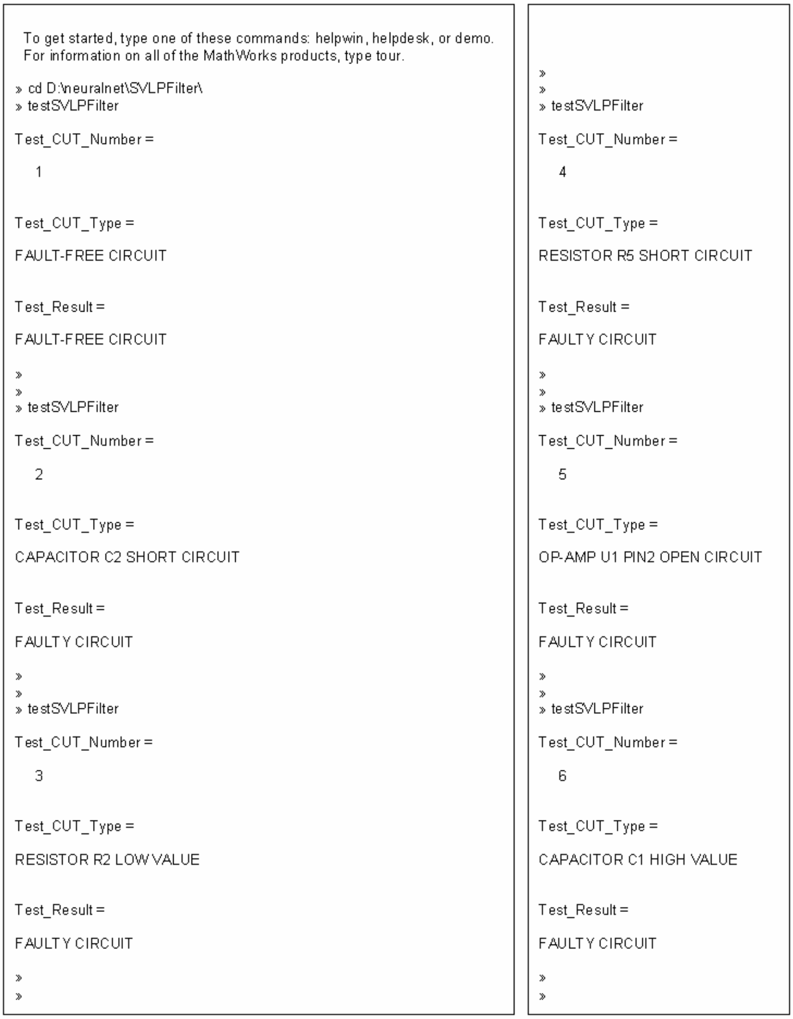


***Figure 4.24:** The nominal output response, the extreme tolerance output response, the nominal ANN model output, the extreme tolerance ANN model output, the nominal error output, and the extreme tolerance error output of the fault-free state variable filter (Fig. 4.22).*

To test an actual filter circuit using this technique the ANN model of the circuit is first created using the output of the circuit under test as the target vector. The output of this model is then compared to the output of the nominal fault-free ANN model to generate the residual (signature). This signature is now compared with the fault-free signature to detect fault in the circuit, if any. Figure 4.25 gives session results of the go/no-go testing algorithm developed and stored in a MATLAB® m-file named testSVLPFilter for this purpose.

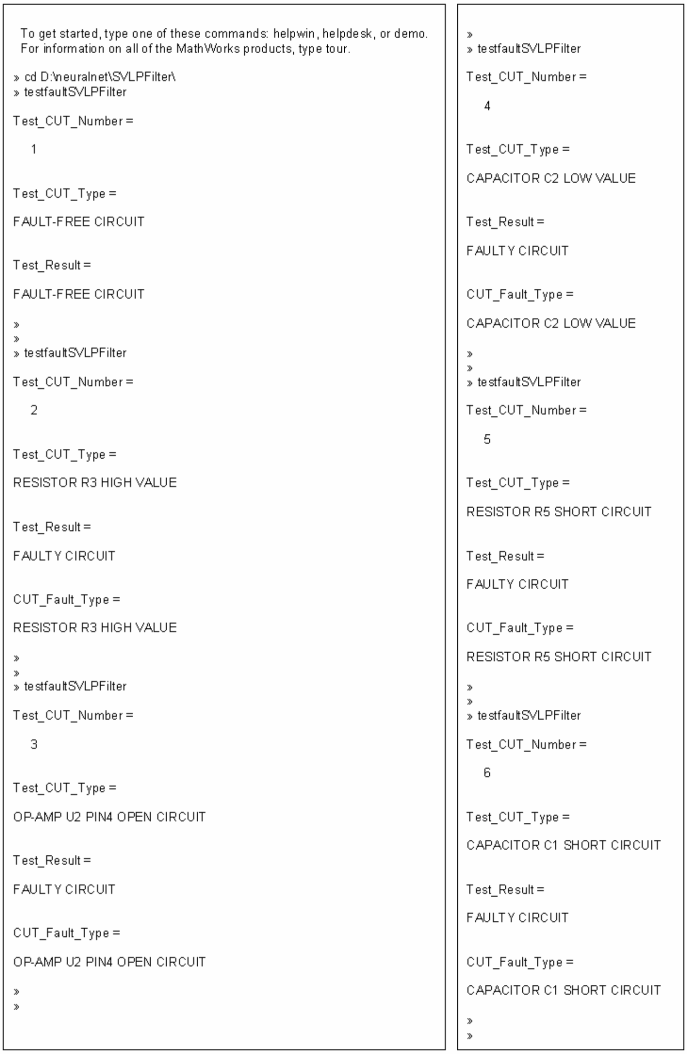


```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SVLPFilter\
» testSVLPFilter

Test_CUT_Number =

     1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testSVLPFilter

Test_CUT_Number =

     2


Test_CUT_Type =

CAPACITOR C2 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testSVLPFilter

Test_CUT_Number =

     3


Test_CUT_Type =

RESISTOR R2 LOW VALUE


Test_Result =

FAULTY CIRCUIT

»
»
```

```
»
»
» testSVLPFilter

Test_CUT_Number =

     4


Test_CUT_Type =

RESISTOR R5 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testSVLPFilter

Test_CUT_Number =

     5


Test_CUT_Type =

OP-AMP U1 PIN2 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testSVLPFilter

Test_CUT_Number =

     6


Test_CUT_Type =

CAPACITOR C1 HIGH VALUE


Test_Result =

FAULTY CIRCUIT

»
»
```

Figure 4.25:** **A session result of the go/no-go testing algorithm (testSVLPFilter.m) for the state variable filter.

If a fault is detected, the cross-correlation coefficient of this fault signature with the nominal fault-free signature is evaluated and compared with that of the individual fault signatures with the nominal fault-free signature to identify the type of fault uniquely. A fault is classified to belong to a particular type if the cross-correlation coefficient of the observed fault signature with the nominal fault-free signature falls within the band of nominal and extreme tolerance values of the cross-correlation coefficients of a particular type of fault signature with the nominal fault-free signature. Some session results obtained from the fault identification algorithm written and stored in a MATLAB m-file named testfaultSVLPFilter are shown in Figure 4.26. Examples of the nominal output response, the extreme tolerance output response, the nominal ANN model response, the extreme tolerance ANN model response, the nominal error output, and the extreme tolerance error output of a few faulted state variable filter circuits are depicted in Figures 4.27.



```
  To get started, type one of these commands: helpwin, helpdesk, or demo.
  For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SVLPFilter\
» testfaultSVLPFilter

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testfaultSVLPFilter

Test_CUT_Number =

    2


Test_CUT_Type =

RESISTOR R3 HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R3 HIGH VALUE

»
»
» testfaultSVLPFilter

Test_CUT_Number =

    3


Test_CUT_Type =

OP-AMP U2 PIN4 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

OP-AMP U2 PIN4 OPEN CIRCUIT

»
»
```

```
»
» testfaultSVLPFilter

Test_CUT_Number =

    4


Test_CUT_Type =

CAPACITOR C2 LOW VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C2 LOW VALUE

»
»
» testfaultSVLPFilter

Test_CUT_Number =

    5


Test_CUT_Type =

RESISTOR R5 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R5 SHORT CIRCUIT

»
»
» testfaultSVLPFilter

Test_CUT_Number =

    6


Test_CUT_Type =

CAPACITOR C1 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C1 SHORT CIRCUIT

»
»
```

Figure 4.26: A session result of the fault classification algorithm (testfaultSVLPFilter.m) for the state variable filter.

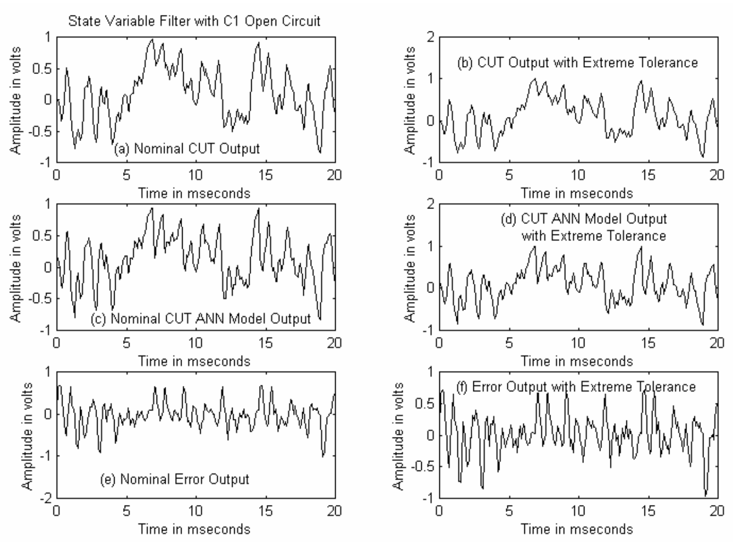


Figure 4.27: ***(i) The nominal output response, the extreme tolerance output response, the nominal ANN model, the extreme tolerance ANN model, the nominal error output, and the extreme tolerance error output of the state variable filter circuit using (Fig. 4.22) with C1 open circuit.***

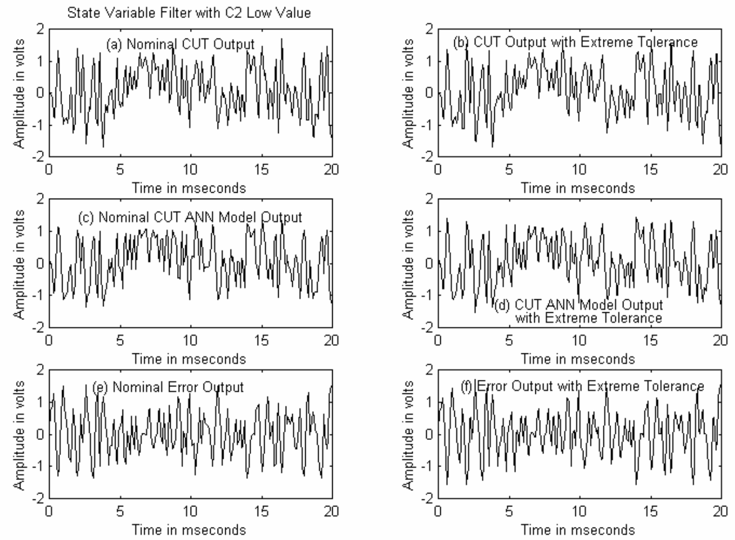


Figure 4.27: ***(ii) The nominal output response, the extreme tolerance output response, the nominal ANN model, the extreme tolerance ANN model, the nominal error output, and the extreme tolerance error output of the state variable filter circuit using (Fig. 4.22) with C2 low value.***

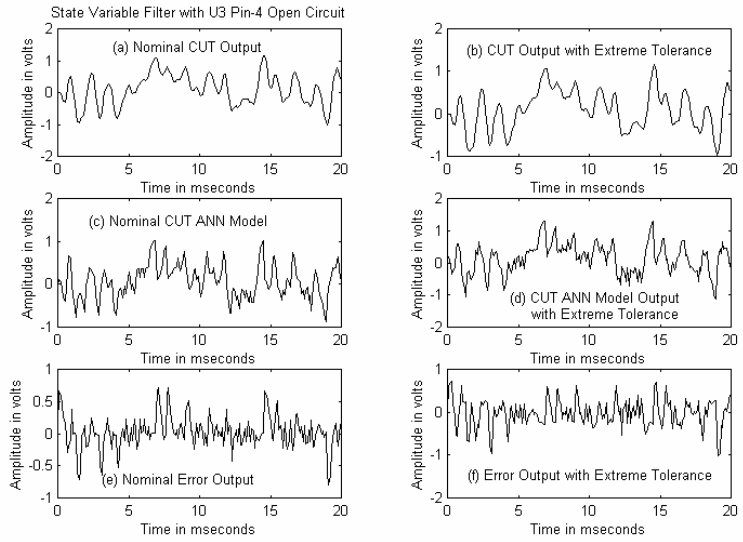


Figure 4.27: ***(iii) The nominal output response, the extreme tolerance output response, the nominal ANN model, the extreme tolerance ANN model, the nominal error output, and the extreme tolerance error output of the state variable filter circuit using (Fig. 4.22) with Op-Amp U3 pin 4 open circuit.***

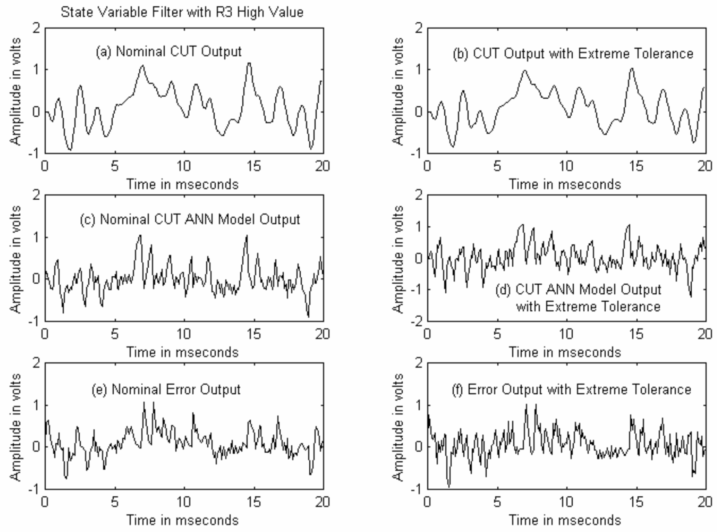


Figure 4.27: ***(iv) The nominal output response, the extreme tolerance output response, the nominal ANN model, the extreme tolerance ANN model, the nominal error output, and the extreme tolerance error output of the state variable filter circuit using (Fig. 4.22) with resistor R3 high value.***

4.6 Experimental verification of the model based observer technique

Analog circuits have traditionally been tested for specifications (e.g. AC gain over a range of frequencies, offset voltage, phase, common-mode-rejection ratio, signal-to-noise ratio, slew rate, input and output impedances, linearity, etc.). This is in part due to the lack of simple fault models in the analog domain. This is an area of ongoing research since functional testing does not provide a quantitative measure of test effectiveness (fault coverage) and it usually results in longer test times because of redundant testing. The introduction of the *stuck-at* fault model for digital circuits enabled digital testing to cope with the exponential growth in the digital circuit size and complexity. Indeed, the *stuck-at* fault model enabled the functional testing to be replaced by structural testing and acted as a measure to quantify the quality of the test plan, permitting test requirements and benchmarking of design-for-test (DfT) strategies. Hard fault modelling and simulation of analog circuits has been suggested in Chapter 2 (Section 2.3), where it may be noted that the faulty analog behaviour is modelled as a modification of the nominal macro-model of a device (component). These fault models give the basis for replacing the conventional functional testing in analog circuits by the universal structural testing, a requirement for incorporation of DfT in analog circuits.

The analysis of the output response for analog circuit is complicated by the fact that analog signals are inherently imprecise. Given a specified input stimulus, the analysis of the output response must take into account the following facts:

- the response can be imprecise due to noise and tolerance specifications of the circuit that generates the waveform, and
- parametric tolerance of the circuit under test that will cause deviations from the ideal response.

Matching the outputs of two identical circuits can serve the purpose of analysis. When identical outputs are not available the approach of generation of a signature that describes the waveform of the output response can be used. Experimental methods and results are presented to illustrate the validity of the methodology described in Section 4.5.

Figure 4.28 gives the test set-up used in the experimentation. Following equipments were used:

1. Workstation to control the arbitrary signal generator and carry out all data processing work,
2. Arbitrary waveform generator, model 33250A, Agilent Technologies,
3. DC power supply, model E3630A, Agilent Technologies, and

4. Mixed-signal oscilloscope, model 54624A, Agilent Technologies.

The test circuit of Figure 4.16 (single amplifier band-pass filter circuit using µA 741 Op-Amp) was assembled on a PCB and connected as shown in Figure 4.28. The arbitrary waveform generator was programmed through the workstation to deliver a pseudo random noise (PRN) by inputting amplitude values generated by a program in MATLAB®.

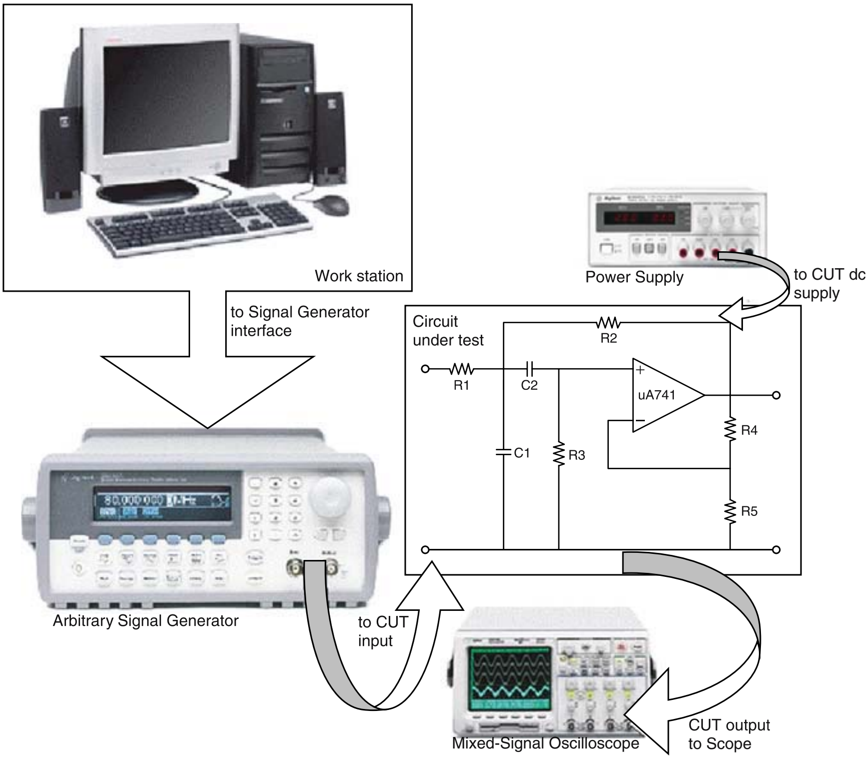


Figure 4.28: Experimental test set-up.

The response of the filter circuit to this PRN was measured and stored through the mixed-signal oscilloscope that has the facility to store the screen displayed data in CSV (comma separated variable) format in a floppy diskette. The data so stored could be read later through Microsoft Excel for further processing in the fault detection and diagnosis. Catastrophic (short/open) as well as parametric fault models were considered in the experiment: two catastrophic faults (short and open) and two parametric faults (high and low value) for a resistor or capacitor and five catastrophic faults for the Op-Amp were studied. Each short fault was modelled by placing a resistor of 1Ω in parallel with the faulty component and each open fault was modelled by inserting a resistor of 1MΩ in series with the faulty component. Similarly, parametric faults (high and low value) were modelled by replacing a particular component by another component of high/low value. The component to be faulted was changed to 6σ more or less the nominal value to induce high or low value fault, respectively, during the experimentation. Based on the fault model, 19 hard faults and 14 soft faults were tested.

The PRN output from the arbitrary signal generator (as programmed through the workstation) and the actual output of the fault-free single amplifier band-pass filter circuit subject to this PRN input are shown in Figure 4.29.

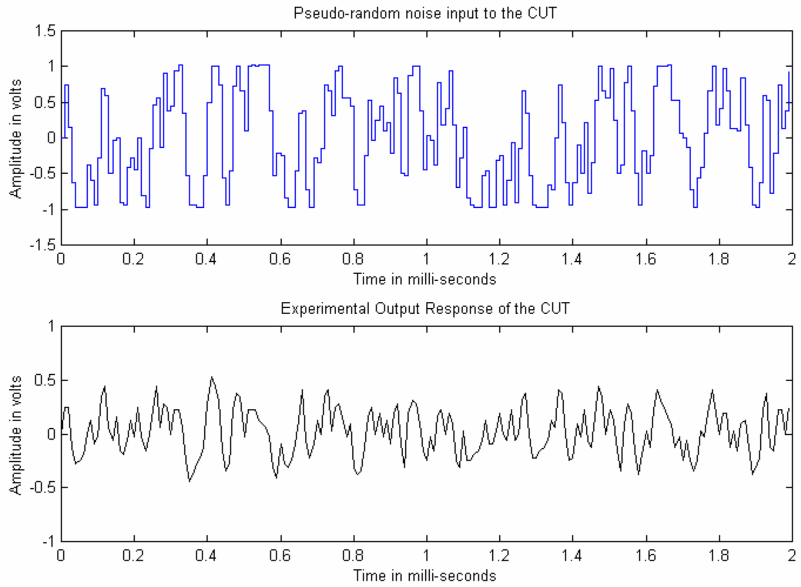


Figure 4.29: Experimental pseudo random noise input and output waveforms of the single amplifier band-pass filter (Fig. 4.16).

The input and output waveform data of Figure 4.29 were stored in a floppy diskette and transferred to MATLAB®. Artificial neural network (ANN) model for the observed circuit under test (CUT) was then simulated using the neural network toolbox. The nominal model of the CUT was obtained from PSpice simulations (Section 4.5.2.1). The output of the simulated nominal CUT, experimental output of the actual CUT, output of the nominal ANN model, output of the actual ANN model and the simulated and experimental residual (error output) of the fault-free single amplifier band-pass filter are shown in Figure 4.30.

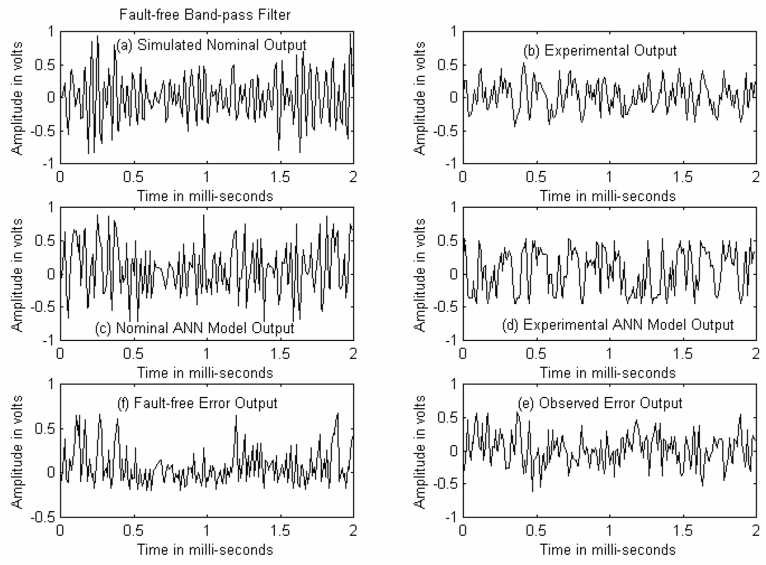


***Figure 4.30:** Output of the simulated nominal CUT, experimental output of the actual CUT, output of the nominal ANN model, output of the actual ANN model and the simulated and experimental residual (error output) of the fault-free single amplifier band-pass filter (Fig. 4.16).*

Different types of hard (open and short) and soft (variations in the component parameter values) faults were introduced. The observed output waveform data were stored in a floppy diskette for use in MATLAB® for training and simulation of the ANN models of these faulty circuits. Output responses of the simulated CUT, experimental output responses of the actual CUT, outputs of the simulated ANN models, outputs of the actual ANN model and the simulated and experimental residuals (error outputs) of some of the faulty single amplifier band-pass filter circuits are shown in Figures 4.31.

Please refer to Section 4.5.2.1 to test the validity of the fault detection and identification scheme. A fault was introduced in the experimental circuit at random. This faulty circuit output response to the PRN was observed and stored in a floppy diskette. The data was then transferred to MATLAB® and an ANN model of the faulty CUT was simulated. The output response of this ANN model was compared with the output response of the nominal fault-free ANN model obtained through earlier simulation to generate the residual. Both the models were simulated with the same PRN as their input stimuli.

The residual so generated is now compared to the fault–free residual stored in the model bank from earlier simulations using the go/no-go algorithm stored in the MATLAB® m-file testSABPFilter. If a fault is detected, the

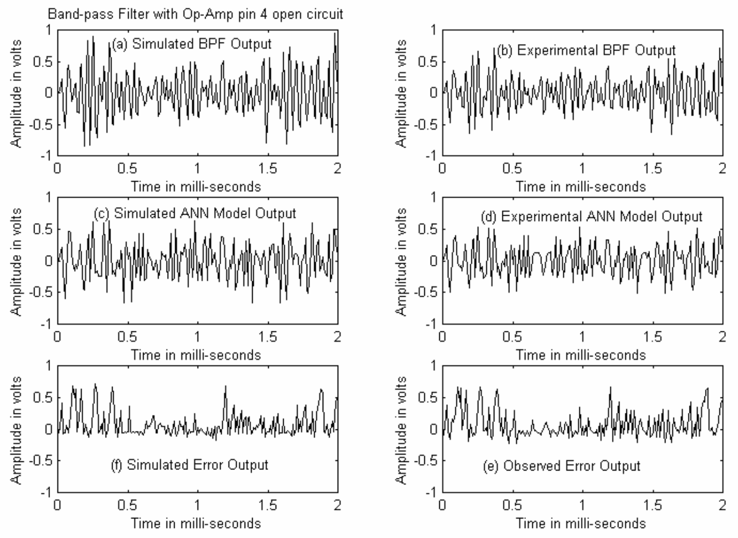


Figure 4.31: ***(i) Output of the simulated nominal CUT, experimental output of the actual CUT, output of the nominal ANN model, output of the actual ANN model and the simulated and experimental residual (error output) of the faulty single amplifier band-pass filter (Fig. 4.16) with Op-Amp pin 4 open circuit.***

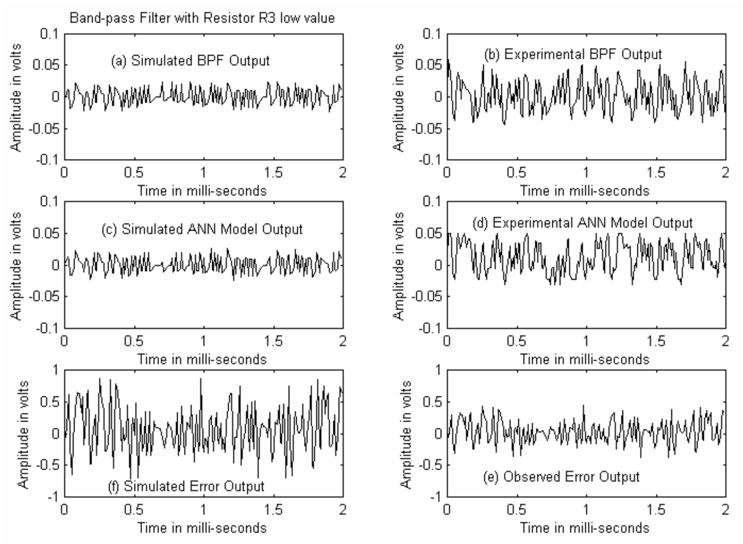
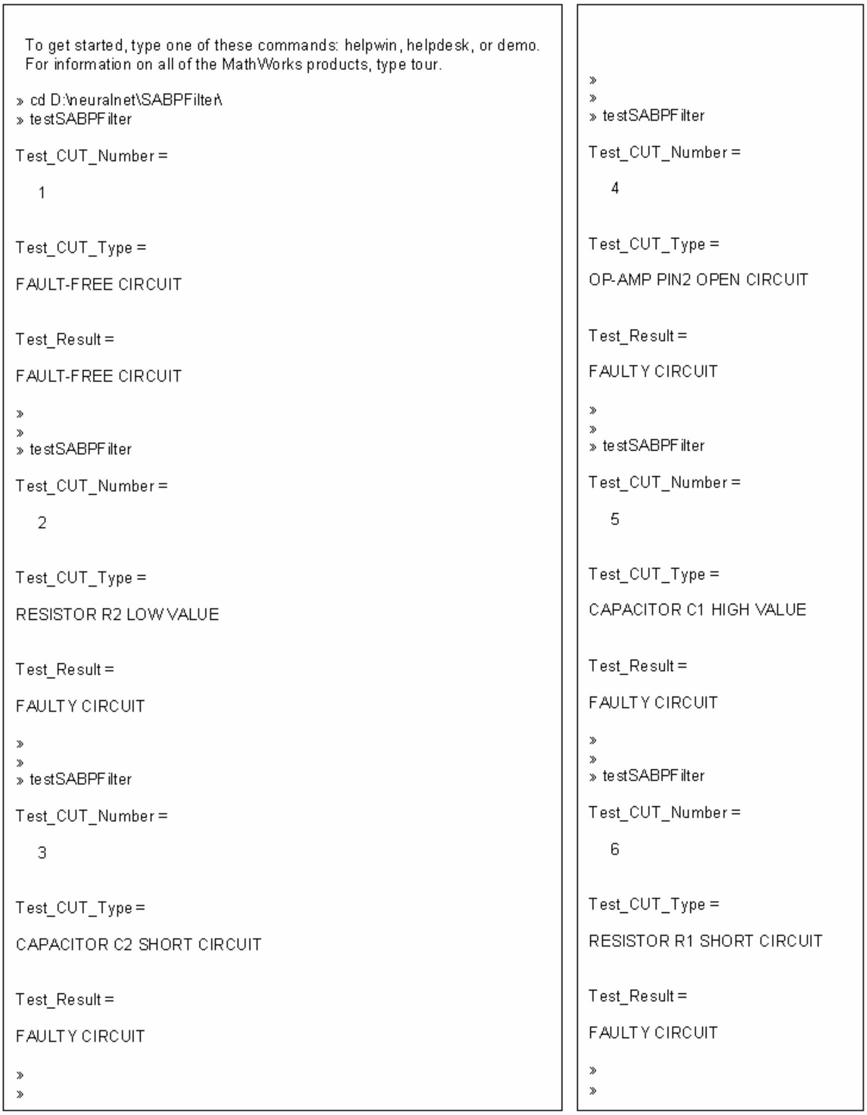


Figure 4.31: (ii) Output of the simulated nominal CUT, experimental output of the actual CUT, output of the nominal ANN model, output of the actual ANN model and the simulated and experimental residual (error output) of the faulty single amplifier band-pass filter (Fig. 4.16) with resistor R3 low value.

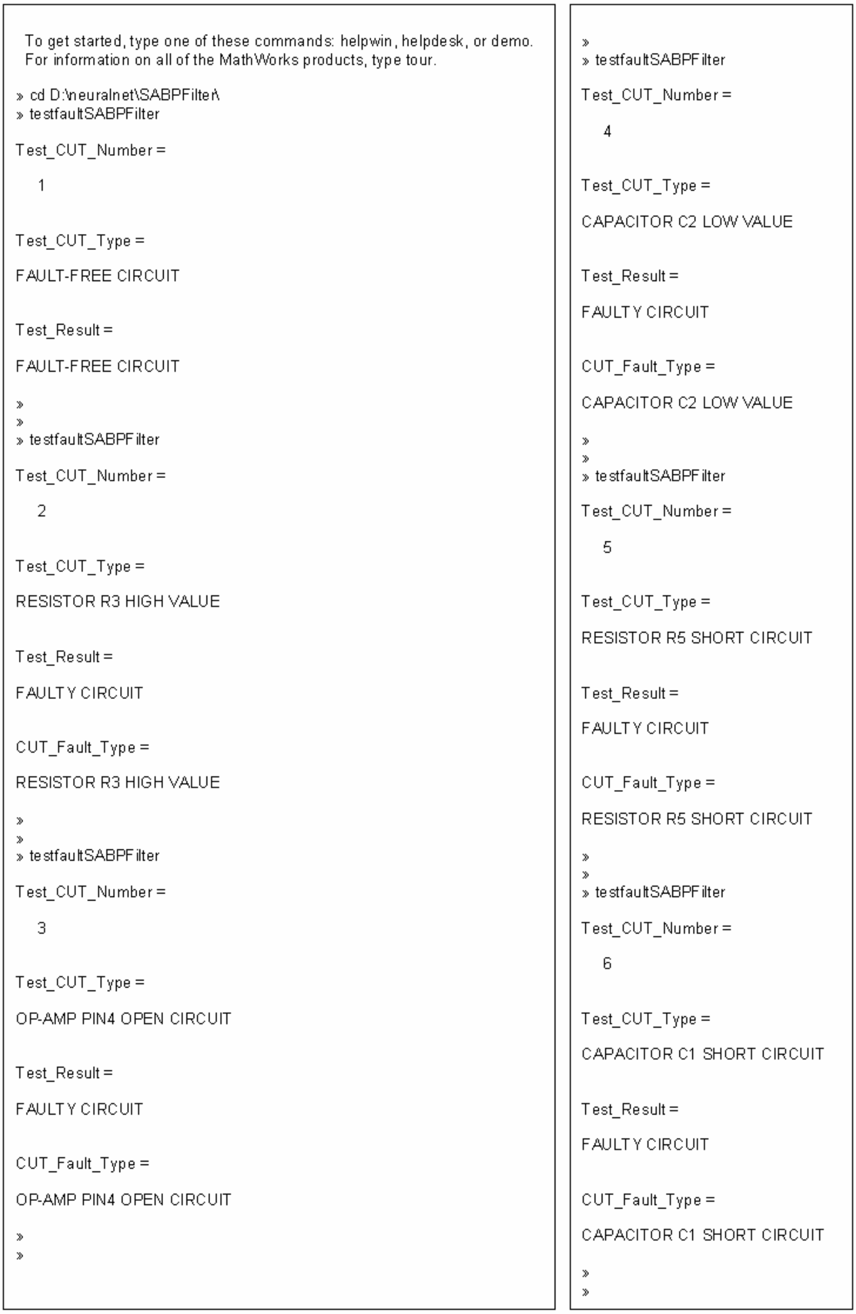
fault detection and diagnosis algorithm stored in the MATLAB® m-file testfaultSABPFilter, is used to identify the type of fault. Some of the session results obtained from the go/no-go algorithm are shown in Figure 4.32 while those for the fault detection and diagnosis algorithm are given in Figure 4.33.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SABPFilter\
» testSABPFilter

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    2


Test_CUT_Type =

RESISTOR R2 LOW VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    3


Test_CUT_Type =

CAPACITOR C2 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
```

```
»
»
» testSABPFilter

Test_CUT_Number =

    4


Test_CUT_Type =

OP-AMP PIN2 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    5


Test_CUT_Type =

CAPACITOR C1 HIGH VALUE


Test_Result =

FAULTY CIRCUIT

»
»
» testSABPFilter

Test_CUT_Number =

    6


Test_CUT_Type =

RESISTOR R1 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT

»
»
```

Figure 4.32: A session result of the go/no-go testing algorithm (testSABPFilter.m) for the single amplifier band-pass filter.



```
 To get started, type one of these commands: helpwin, helpdesk, or demo.
 For information on all of the MathWorks products, type tour.

» cd D:\neuralnet\SABPFilter\
» testfaultSABPFilter

Test_CUT_Number =

    1


Test_CUT_Type =

FAULT-FREE CIRCUIT


Test_Result =

FAULT-FREE CIRCUIT

»
»
» testfaultSABPFilter

Test_CUT_Number =

    2


Test_CUT_Type =

RESISTOR R3 HIGH VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R3 HIGH VALUE

»
»
» testfaultSABPFilter

Test_CUT_Number =

    3


Test_CUT_Type =

OP-AMP PIN4 OPEN CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

OP-AMP PIN4 OPEN CIRCUIT

»
»
```

```
»
» testfaultSABPFilter

Test_CUT_Number =

    4


Test_CUT_Type =

CAPACITOR C2 LOW VALUE


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C2 LOW VALUE

»
»
» testfaultSABPFilter

Test_CUT_Number =

    5


Test_CUT_Type =

RESISTOR R5 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

RESISTOR R5 SHORT CIRCUIT

»
»
» testfaultSABPFilter

Test_CUT_Number =

    6


Test_CUT_Type =

CAPACITOR C1 SHORT CIRCUIT


Test_Result =

FAULTY CIRCUIT


CUT_Fault_Type =

CAPACITOR C1 SHORT CIRCUIT

»
»
```

Figure 4.33: A session result of the fault identification algorithm (testfaultSABPFilter.m) for the single amplifier band-pass filter.

4.7 Summary

This chapter has presented the methodology of fault diagnosis. The general approach to fault diagnosis procedure has been discussed. Some prominent fault diagnosis techniques are presented in this chapter. The fault dictionary approach is suitable for detecting hard and soft faults. The fault dictionary approach consist of two stages namely, pre-test analysis and post test analysis. With DSP based techniques AC measurement can be performed in a very short time. It is more accurate than analog only testing. Despite its main advantages DSP based technique expects the knowledge of physical and mathematical principles of each test from the test engineer. The procedure is more complicated than the conventional testing by bench equipment.

Section 4.5 provides a basic foundation for model based observer technique of fault diagnosis of analog integrated circuits, even though other methods/techniques we have discussed are also in use. Fault diagnosis of stand alone analog integrated circuit has been discussed in subsection 4.5.1. As examples A 741 operational amplifier and MOS operational amplifier are considered in the sub-subsections 4.5.1.1 and 4.5.1.2 respectively. Fault diagnoses of analog filters have been discussed in sub subsections 4.5.2.1 and 4.5.2.2. An experimental set up for fault detection and diagnosis has been given. The model based observer technique is thus expected to be useful in the fault diagnosis of various analog integrated circuits, some of which have been illustrated here.

Exercises

4.1 Prepare fault list for the μA 741 of Fig. 4.6. There will be 208 hard and 336 soft faults altogether.

4.2 What do you understand by the terms: *fault dictionary, residuals, Monte-Carlo technique and correlation coefficient?* Discuss how those are relevant in analog circuit fault diagnosis methodology.

4.3 What DSP based techniques achieve by elaborate programming, the same is achieved in model based observer technique by use of multiple models. Indicate the probable areas of application of either technique.

4.4 While *go / no-go test* is preferred in the production line testing of analog ICs, subsequent *fault classification* algorithm is more suited for system-on-chips (SOC) testing. Do you agree / disagree? Give reasons.

References

[1] A. Birolini, Quality and Reliability of Technical Systems, Springer-Verlag, Berlin, 1994.

[2] M.J. Ohletz, “Hybrid built-in self-test (HBIST) for mixed analogue/digital integrated circuits”, *Proceedings, 2nd European Test Conference*, pp. 307-316, April 1991.

[3] W. Hochwald and J.D. Bastian, “A dc approach for analog fault dictionary determination”, *IEEE Transactions on Circuits and systems*, Vol. CAS-26, pp. 523-529, July 1979.

[4] M.V. Mahony, DSP-Based Testing of Analog and Mixed-Signal Circuits, IEEE Computer Society Press, 1987.

[5] M. F. Toner and G. W. Roberts, “A BIST Scheme for an SNR, gain tracking, and frequency response test of a sigma- delta ADC”, IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing, pp. 1-15, Jan. 1995.

[6] C-Y. Pan and K-T. Cheng, “Pseudo-random testing and signature analysis for mixed-signal circuits”*, Digest of Papers*, *IEEE/ACM International Conference on Computer–Aided Design* (ICCAD-95), pp. 102-107, November 1995.

[7] C-Y. Pan, K-T. Cheng and S. Gupta, “Fault macro-modeling and a test strategy for Opamps”, *Journal of Electronic Testing: Theory and Application*, Vol. 9, pp. 225-235, December 1996.

[8] R.M. Gray and J.W. Goodman, Fourier Transforms; An Introduction for Engineers, Kluwer Academic Publishers, Norwell, Massachusetts, 1995.

[9] M.V. Mahony, “New techniques for high speed analog testing”, *Proceedings, International Test Conference*, pp. 589-597, October 1983.

[10] R. Patton, P. Frank and R. Clark, eds., Fault Diagnosis in Dynamic Systems: Theory and Applications, Prentice Hall International (UK) Ltd., Hertfordshire, 1989.

[11] R. Isermann, “Process fault detection based on modeling and estimation methods - a survey”, *Automatica*, Vol. 20, pp. 387-404, July 1984.

[12] P.M. Frank, “Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey and some new results”, *Automatica*, Vol. 26, pp. 459-474, May 1990.

[13] P. Kabisatpathy, A. Barua and S. Sinha, “Fault detection and diagnosis in analog integrated circuits using artificial neural networks in a pseudorandom testing scheme”, *Proceedings,3rd Intrnational Conference on Electrical & Computer Engineering* (ICECE 2004), pp. 52-55, December 2004.

[14] B. Kaminska, K. Arabi, I. Bell, J. L. Heurtas, B. Kim, A. Rueda and M. Soma, “Analog and mixed-signal benchmark circuits – first release”, *Proceedings, 1997 IEEE International Test Conference* (ITC’97)*,* pp. 183-190, November 1997.

[15] B.A. Wooley, S J. Wong and D O. Pederson, “A computer-aided evaluation of the 741 amplifier”, *IEEE Journal of Solid-State Circuits*, Vol. SC-6, No. 6, pp. 357-366, December 1971.

[16] P.W. Tuinenga, SPICE: A Guide to Circuit Simulation and Analysis using PSpice, Prentice-Hall, New Jersey, 1992.

[17] H. Demuth and M. Beale, Neural Network Toolbox User’s Guide, The MathWorks, Inc., Natick, 1994

[18] IEEE mixed-signal benchmark circuit home page available on the web-page at: http://faculty.washington.edu/mani soma/madtest/benchmarks/OpAmp.htm.

[19] R. Spina and S.J. Upadhyaya, "Linear circuit fault diagnosis using neuromorphic analyzers", IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 44, pp. 188-196, March 1997.

Chapter 5

DESIGN FOR TESTABILITY AND BUILT-IN SELF-TEST

5.1 Introduction

In recent years, the growing importance of analog circuitry in commercial applications like mobile communications, process control, automotive systems application specific integrated circuits (ASICs) and smart sensors has prompted the integrated circuit (IC) designers to implement both analog and digital components to form an entire complex system-on-chip (SOC). Consequently, testing is becoming one of the major cost factors in the overall IC manufacturing expenses. Cost as well as quality is crucial in many of these devices as most markets are characterised by high volume, low cost and high quality. The analog tests are expensive in terms of both development costs and implementation [1]. Although the analog circuitry generally comprises, only about 10% of the total chip area, it covers the major part of the testing cost. So, a reduction in the cost of the analog testing will be directly profitable in terms of production of the SOCs. For the manufacturers to deliver quality products in reasonable time, extensive testing and fault diagnosis techniques have to be developed and implemented for the analog portion of the SOCs [2].

Functional test techniques for digital ICs were proved unrealistic many years ago due to the escalating size of the test sets and associated time required to run the test vectors. Alternative techniques have since been established; testing for permanent structural faults (stuck-at, bridging, stuck-open), I_{DDQ} testing and delay testing. Some of the techniques are compatible with inductive fault analysis (IFA) based test approaches, where the device under test (DUT) has to be proven structurally sound. Also in the digital domain, design for testability (DfT) is well established [3], with full or partial scan [4] being implemented successfully in the majority of complex products. In addition, the IEEE standard 1149.1 test access port and boundary scan architecture [5] has been well accepted by digital designers. Due to the dramatic increase in complexity of digital circuits, built-in self-test (BIST) facility is now receiving great interest and has been realised mainly in highly structured commercial designs to implement some of the test functions on-chip.

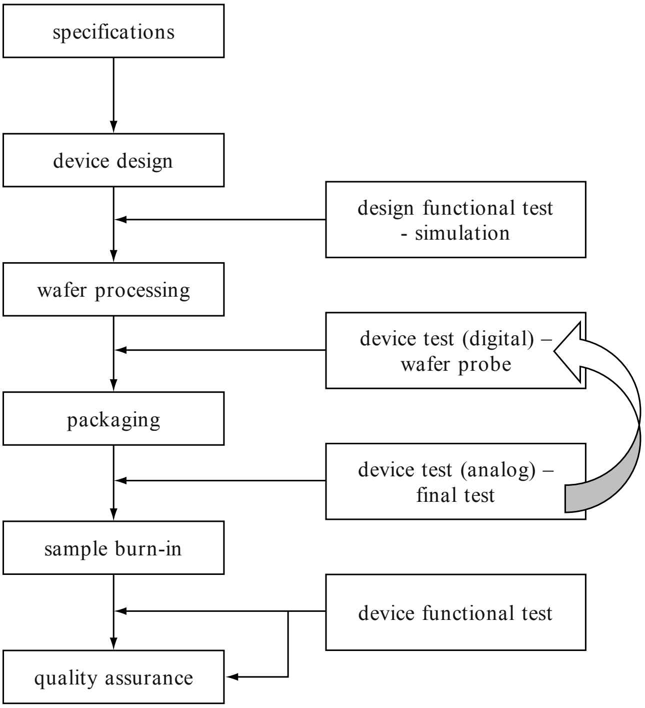


Figure 5.1: ***Simplified IC manufacturing flow diagram.***

In analog and mixed-signal circuits, functional testing is still the norm for both wafer level and packaged tests. Essentially the goal is to verify each IC against critical specifications [6][7]. Figure 5.1 illustrates where testing is applied in the manufacturing process of a mixed-signal SOC. The optimisation of such circuit specific test programs is difficult to handle and expensive due to the required engineering resources. Additionally, neither the test quality nor the yield can be estimated, as the product is not directly tested for defects. Usually, the IC designer receives the circuit specifications and uses computer-aided-design (CAD) tools to create and verify a new cell design and/or composes and verifies the system. When the design creation itself has been completed, the test program generation begins. The generated test program is then applied to test the device under test (DUT) at the wafer level (or pre-test) and/or when the device has been packaged. The earliest feedback can theoretically be provided from test failure analysis. Critical feedback may also be provided by the customer who will identify shipped failures however, at this stage, returns greater than several hundred ppm is likely to be a serious issue in terms of cost to and reputation of the supplier. If the product life is longer than one technology generation or requires several production batches spread over several years, improvements in the test program itself may solve the problem and increase the test coverage. In the worst case, the cell or system may have to be partially modified. In general, the feedback is provided far too late, and test generation is not synchronised with design creation.

To synchronise test generation and design creation, enabling early feedback, provide early test coverage figures and address testability problems in the early design stages, the use of a computer-aided test (CAT) tool is essential. Within this simulation environment, which is ideally integrated into the CAD tool, the design and/or test engineer is able to compute coverage figures for particular test programs. A fault and defect extraction facility and fault (simulation) models are needed as inputs to compute the test coverage for a simulated test strategy. With increasing use of the CAT and CAD environment and experience, the designer now has the ability to develop the test plan early on in the design cycle and to find the best compromise between functionality, performance and test. This new paradigm has come to be known as Design-for-Testability (DfT).

It is interesting to note that in the 1997 SIA roadmap [8], mixed signal test is given a low profile relative to the need for digital BIST techniques to address the increased use of intellectual property, the emergence of SOC and the growing importance of covering parametric and delay faults and verifying signal integrity. It is however acknowledged that the inevitable increase in speed and integration of RF and mixed signal functionality is a serious issue for test in the generic context. The experience in many companies however is that the implementation of analog test programs although being quite feasible provided adequate controllability is available, has serious problems due to high automated test equipment (ATE) and test time costs, lack of fault coverage metric and unacceptable test development overhead relative to digital functions. These issues must be solved and DfT solutions appear to be the way forward. In this context DfT can be interpreted as:

"Any design modification or constraint that can improve test access, reduce test cost, improve test efficiency or improve fault coverage".

There are however a number of hurdles to cross before analog DfT solutions are feasible:

- DfT solutions may involve additional circuitry that may degrade primary circuit specifications, increase the power consumption and die size.
- DfT solutions may involve the need to implement a new design strategy.
- DfT solutions may need additional analog functions hence eliminating simple implementations.
- DfT solutions involving BIST or on-chip processing may not always take into account the problems of noise, process variations and measurement inaccuracies that are inherent to analog functions.

Although the growth in the mixed signal ASIC market for process control, transportation and instrumentation applications is considerable, many of these applications are safety critical, or demand high dependability, or high reliability. Solving these requirements inevitably may lead to an increased use of on-line functional verification to allow safe, dependable and reliable operation. Furthermore, legislative requirements and demands from insurers in many cases necessitate proof on "completeness" in testing after manufacture and even on-line verification capabilities for certain applications [9].

5.2 Design-for-testability (DfT) approaches

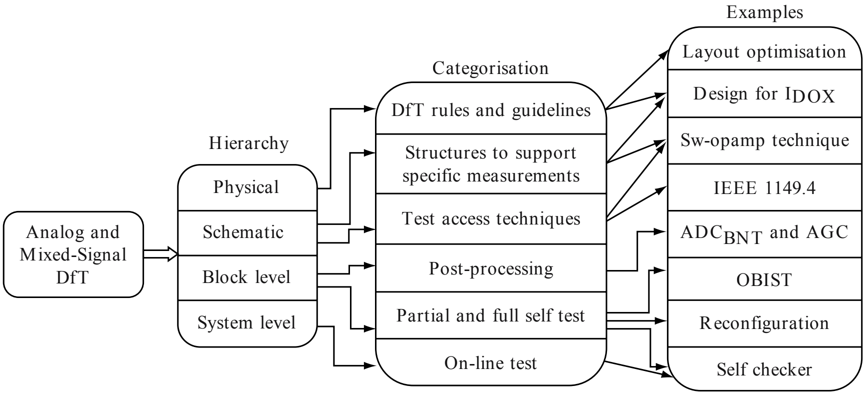


Figure 5.2: ***Analog and Mixed-signal DfT options with examples.***

Figure 5.2 attempts to summarise the analog and mixed-signal DfT field with a number of examples. Despite the differences in their approach they all have the same aim of improving the testability of IC designs. The following review by no means covers all solutions but rather attempts to identify representative examples that have a significant chance of helping to meet test objectives.

5.2.1 Rules and guidelines

Most DfT rules and guidelines to improve testability are 'good practice' or 'common sense' knowledge on circuit design gathered from previous experience [10]. A set of general principles has been developed [11], quite similar to the 'ad-hoc' testing techniques in digital design:

- Partition the circuit in macro blocks;
- Control macro inputs;
- Observe macro outputs;
- Disable feedback paths;
- Place digital storage elements at A/D and D/A interfaces;
- Use digital test access port to select test mode

These guidelines can also be integrated in CAD tools and monitored via design rule checkers. In addition, constraints on the physical design process for improved testability are essentially DfT techniques. Prieto [12] has proposed an interesting method for influencing the layout process to reduce fault probability and hence improve testability. The technique involves the use of Inductive Fault Analysis to generate a relationship between fault probability for specific devices and parameters of that device such as width, length and number of series connected devices. Once this relationship is known, it is possible to extract a fault probability for a complete design and more importantly, a utility can be built into an automatic layout tool that can calculate the fault probability of a certain layout solution. The main application of this technique is the automatic generation of defect tolerant layouts, however, the probability functions for each structure on the device must first be calculated for each technology used.

Prieto's technique can also be used to generate a fault list and if combined with a fault simulation process, it is possible to back annotate and change layouts to eliminate specific difficulties to detect faults. This has in fact been done by Atzema [13] and Lechner [14] although both of these layout modifications have been identified through fault simulation against layout extracted fault lists rather than using Prieto's method. In [13], a solution that involves reordering two bit lines of a bus that are used to program the pole frequency of a switched capacitor filter for an audio application eliminates an undetectable fault. In [14], a high probability of parametric faults in a resistor ladder was found to eliminate the possibility of reducing the test program complexity. This led to a revised layout allowing a significant reduction in test time.

5.2.2 Support for external test

Power supply current monitoring or I_{DDQ} testing [15] has become a standard test technique in industry for digital designs. Its application to mixed signal and analog functions is now an active research topic. The utilisation of the technique requires careful circuit partitioning strategies that involve, at the minimum, separate power supplies and in many cases the integration of test modes (where for example analog modules are set into a low power state) to ensure the analog circuitry has negligible effect on current levels during digital I_{DDQ} test. Various forms of current testing have been investigated for analog circuitry. Process variations in certain analog components, particularly the bias chains can cause wide variations in fault-free current levels that can effectively mask many fault effects [16]. Possible solutions have been proposed including taking two supply current measurements for two “opposite polarity” input signals, and processing the results so that the bias current component is effectively cancelled [17]. It may also be feasible for certain circuits to include additional current mirror structures to effectively mirror the bias current for an analog block to an additional probe pad. This would be a form of DfT or partial replication and could be used to provide access to the bias current for the analog blocks during probe test.

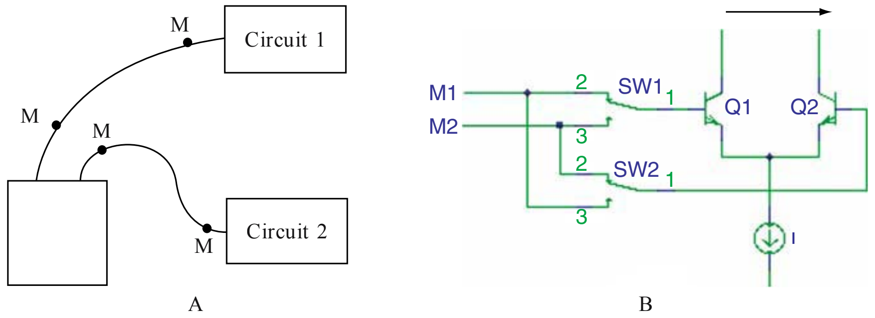


Figure 5.3: ***A. Supply current can be measured through the voltage drop across supply lines. B. A bipolar differential pair that can be used as a V to I converter.***

The methodology developed by Van Lammeran [18] has some unique features. Here, the current in the supply/ground lines to specific functional blocks is monitored by measuring the voltage drop in the supply rails due to the small resistivity of the material (Figure 5.3A). As this is extremely small, a bipolar differential pair is used to convert the voltage drop to a collector current (Figure 5.3B). To eliminate offsets, two measurements are taken with the differential inputs switched around and to eliminate tail currents, the ratio of the currents in the two collectors is used. The effect of wide fault free supply current distributions due to process variation and mismatch is also reduced by using the ratio of results between different blocks rather than absolute values as pass/fail criteria. This technique is referred to as I_{CCQ} and has been implemented for DfT of single chip television IC by Philips using BiCMOS technology.

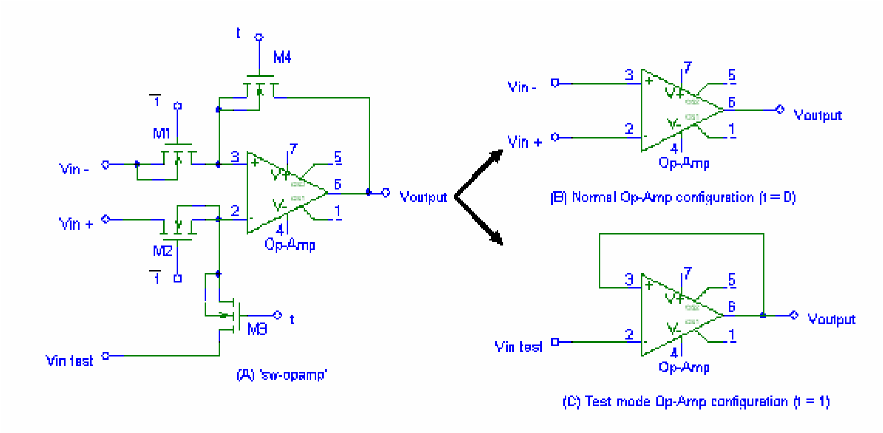
Other 'external' testing techniques in this category are Transient Response Analysis [19] and Residual Multiple Frequency Testing [20]. Transient Response Analysis requires the introduction of a special interface scan structure between analog and digital subsystems. The analog part is excited with impulses and its output is digitised and processed to evaluate the frequency behaviour. Residual Multiple Frequency Testing is suitable for linear analog blocks with bandpass frequency response. Two sinusoidal signals are used, which lie just above and just below the operation bandwidth. By retrieving the amplitudes of test signals at the analog output, faults can be detected. It is worth mentioning that this technique can also be used for concurrent testing, provided that test signals do not interfere with normal operation (i.e., their amplitudes do not saturate the circuit, no harmonic/subharmonic frequencies are generated, and analog output is bandpass filtered to remove test signals).

5.2.3 Access to embedded blocks

Techniques for increasing controllability and observability with adequate resolution are categorised under this section. Scan path [10] and the IEEE 1149.4 standard proposal [21] are examples of structured approaches to access embedded modules or nodes. IEEE 1149.4 is the analog extension of IEEE 1149.1 Digital Boundary Scan architecture. The existing standard proposal is fully compatible with its digital counterpart, thus making possible to test mixed systems with a unified Test Access Port (TAP). The pin overhead is minimum (one analog input and one analog output besides the digital TAP). However, for certain applications where the total pin count is relatively low (especially consumer products) it may not be feasible to assign six or more pins to test.

One possibility to access embedded nodes without the need for analog test pins is to use converters on-chip so that the tester interfaces the circuit under test (CUT) in the digital domain. ADC and DAC in the test circuitry are then required to convert the digital serial I/O stream into analog test signals [22]. One alternative to save the chip area for these converters at the expense of an additional analog test pin are analog shift registers [23].

Besides the requirements to establish a path to embedded nodes, there must be interfaces to inject test signals into the signal path circuitry and the 'sw-opamp' is one solution that causes only minimal degradation of the primary functions [24]. The 'sw-opamp' shown in Figure 5.4 is a reconfigurable op-amp which can either operate in normal mode or in the test mode. By selecting the appropriate 'sw-opamp', it is possible to inject test signals at almost any point in the analog subsystem.



(A) 'sw-opamp'

(B) Normal Op-Amp configuration (t = 0)

(C) Test mode Op-Amp configuration (t = 1)

Figure 5.4: ***The 'sw-opamp' or the reconfigurable operational amplifier.***

Further applications of the 'sw-opamp' have been proposed by D Vazquez et al [25]. Here all operational amplifiers are replaced by 'sw-opamps'. All stages except the stage under test are put into test mode (op-amps in unity gain configuration). The input to the stage under test will then be equal to the primary input and the output of the stage under test will be propagated through subsequent stages to the primary output. This technique has been applied to a second order active RC filter and a second order SC filter. In addition, fully differential versions have been built into a 6th order bandpass filter with minimal performance degradation and silicon overhead [26].

5.2.4 On-chip test evaluation

Off-chip test signal generation and evaluation imposes severe restrictions on the test signal itself. The maximum frequency, the signal amplitude, or the resolution of the test signal are limited by the clock rates of the digital serial chains, converters, or by chain elements in the analog shift registers.

The quality of the test stimuli and response is limited by the links from the ATE to the chip periphery and from the pins to the embedded modules/ nodes.

The alternative is to generate/evaluate (test) signals on-chip. Here, test information from the tester (or higher level system) is decoded/coded and/or decompressed/compressed. This frees the external tester from dealing with 'difficult' analog patterns at the expense of additional on-chip circuit overhead.

The analog fault checker [27] monitors inputs/outputs of fully differential circuits and uses relative tolerances for self-test evaluation. It has a negligible impact on area and circuit performance. Since it monitors the average value of differential signals, its application is limited to differential circuits; moreover, it can not detect 'symmetric' faults (e.g., differential bridging).

The reduction in test time and improvement in test effectiveness does not necessarily require full self test. An example here is a study on an analog front-end circuit by Lechner [14][28]. Here high test costs are due to the need to verify the gain at each of 32 gain steps. Providing that certain layout and schematic DfT rules have been obeyed, it has been shown that a digital post processing block can be added at the block level, implemented in software on an on-chip microprocessor or within a DSP module on an external tester. The approach is shown in Figure 5.5.

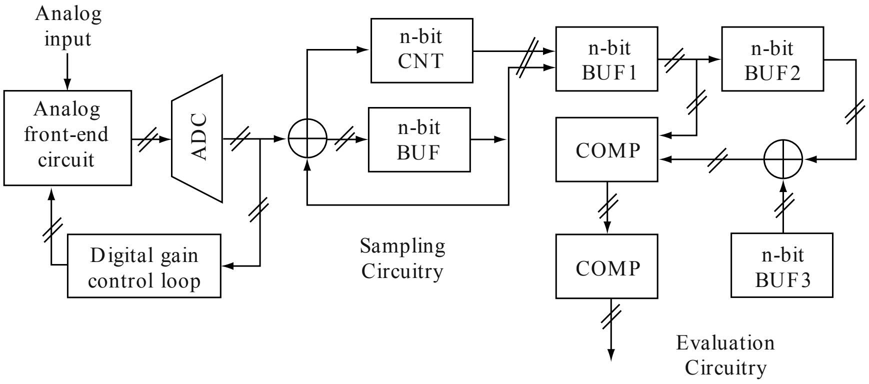


Figure 5.5: ***Test evaluation structure for analog front-end.***

The technique works as follows: An input voltage is fed into the analog front-end circuit and the gain set initialised to its lowest value. The sum of a set number of ADC output patterns is calculated within the *sampling circuitr*y. The output of the sampling block is then latched and the gain set incremented. The value can then be compared against the previous output and the difference can be checked to be larger (or smaller) than a set threshold within the *test evaluation* block. Test parameters can be set by initialising the contents of the registers and the counters. These blocks can be embedded in a digital scan chain that can also be used to read out the test signature.

In addition detailed fault simulation has shown that the majority of faults affecting the analog front-end circuit cause problems with the output voltage swing (OVS). By applying a ramp input to the analog front-end circuit and sampling the output for a fixed gain step, The OVS can be verified. This is achieved by counting the number of samples that are larger than the previous sample by a fixed amount. This algorithm can be implemented in the same digital block. This purely digital circuitry allows the generation of a simple pass/fail output for the gain step size test and the verification of the analog front-end circuit output voltage swing (OVS). Offset effects have been cancelled out, as they influence every sample set in the same way. Only if the offset drives the ADC input out of the valid input range, will the sample sum become faulty (smaller than it should be). However, this would result in a decreased gain step, and also the following gain steps will be too small. By implementing this top-level test evaluation circuitry, the expected test time for not just the analog front-end circuit but the complete chip can be reduced by about 70% without a decrease in test coverage.

5.3 Increased testability with test bus

The primary requirement of any test is the controllability and observability of the states of the device. In digital circuits, this is achieved by the un-intrusive scan-path approach [29] and is practiced widely in the electronics industry. For mixed-signal circuits, analog test buses are commonly used.

5.3.1 Standard test bus

Probably the most widespread method used to gain external access to internal nodes of a mixed-signal circuit is through the use of an analog test bus and analog switches constructed with CMOS transmission gates. When the circuit is in test-mode, the node of interest is connected to the analog test bus where a signal can be brought in or out from off- chip. If two analog buses are used, then a standard input-output measurement can be made. An example of this is illustrated in Figure 5.6. To maximize the dynamic range of the test bus, a fully-differential test approach can also be used.

An obvious drawback to any of these approaches is the need for additional I/O pins, which in many cases may be hard to come by.

Alternatives, like multiplexing I/O pins with the test bus and other parts of the circuit not in use during the test phase is also a possibility. Care must be exercised when on-chip signals are passed through the chip's I/O pins, as the load capacitance on a particular node is now increased significantly. In many cases, signals being driven off-chip should be buffered using an Op-Amp circuit or an equivalent circuit. It is imperative that the inclusion of an analog test bus be considered early on in the design phase to minimize the impact that the test bus will have on the circuit in normal operation.

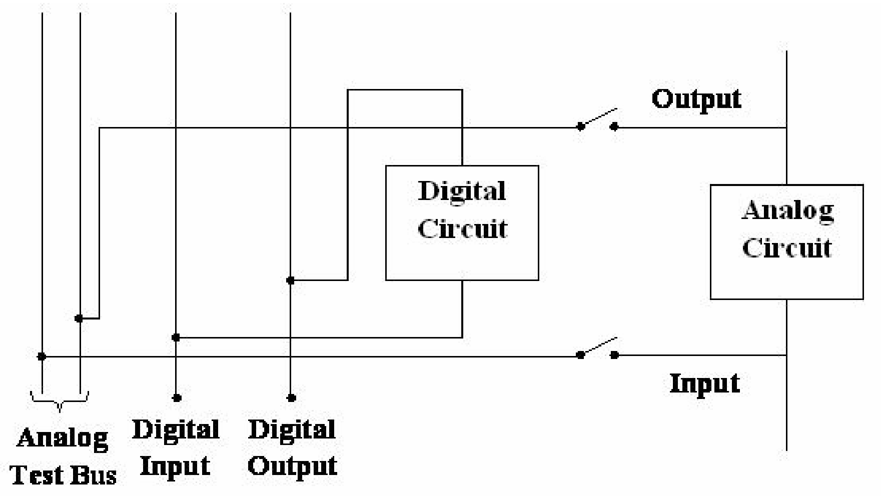


Figure 5.6: ***An analog test bus configuration.***

For very high-frequency narrowband signals, an on-chip wideband under-sampling circuit can be used to mix down the essential information to a lower frequency before being brought off-chip [30]. This eliminates the frequency response limitations of the off-chip driver but imposes very stringent requirements on the timing and control signals. Some form of calibration is also necessary to ensure that the added circuitry is itself functioning within specification. Alternatively, one can make use of the high-speed test bus discussed in the next subsection.

5.3.2 High speed test bus

An interesting proposal for a high-speed test bus for broadband operation is presented in [31]. A node voltage is observed at an external pin by first converting the node voltage into a current using a local V-to-I circuit (in this case a digital tri-state inverter circuit) and then moving it off the chip through the test bus. An external I-to-V circuit is constructed with a high-speed Op-Amp circuit that converts the signal back to its original voltage form for direct access. The test bus is held at a constant voltage level to maximize its bandwidth by eliminating the Miller effect of each inverter. A calibration phase is introduced to determine the correction factors that are needed to compensate for the nonlinear behaviour associated with the inverter circuit. An improvement of 15 dB was reported after calibration. This particular test bus also illustrates the concept of a virtual switch where the on-off action of each switch is embedded into the function of each inverter circuit.

5.3.3 The IEEE 1149.4P test bus

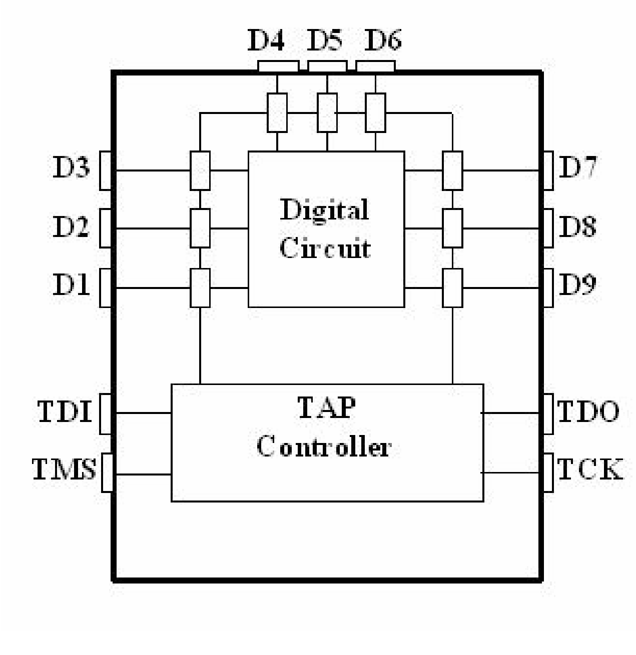


Figure 5.7: An IEEE 1149.1 compliant chip.

Over the past few years a group of international companies and R&D institutions have been working together to define a mixed-signal test bus standard which is compatible with the existing IEEE 1149.1 boundary scan standard. The present day digital test bus standard provides scan path access at the chip I/O boundary allowing individual ICs to be isolated, controlled and observed by user-supplied test vectors through the test access port (TAP) controller as shown in Figure 5.7. Another important aspect of this test bus is it allows for checking of wiring interconnects between neighbouring ICs on the same or different boards. Opens and shorts in a board's wiring interconnect accounts for 80% - 90% of all board failures. Hence the latter aspect of the standard is a significant one. With the introduction of analog components to 1149.1 compliant chips, the ability to isolate faulty interconnects on the analog I/O pins does not exist. Of course, there is also no way of isolating, controlling and observing the behaviour of individual analog components through the TAP controller.

The first milestone of the IEEE 1149.4 working group [32] is to introduce test bus facilities for interconnect tests at the board level. Access to internal analog cores is not an immediate concern, but combining the test bus facilities of the previous section is one possibility. The basic idea of the mixed-signal test bus is the inclusion of a set of digitally-controllable analog boundary cells that can perform the following four functions:

1. disconnect the I/O pin from the analog core,
2. set the I/O pin at a logic high or low level,
3. detect the logic level present on the I/O pin, and
4. connect the I/O pin to a two-wire analog test bus.

The control circuit that performs the above functions is shown in Figure 5.8. The digital control is provided via a slightly modified 1149.1 compatible TAP controller. In test mode, the TAP controller instructs the various boundary cells (both analog and digital) to generate an appropriate logic level that is sent across the interconnect and received by other boundary cells from which a detection routine can deduce the presence of any shorts or opens. Moreover, through application of the analog test bus, an external signal can be injected into different signal paths and picked up on other ones whereby the value of specific external components can be measured.

5.4 Built-in-self-test

The increasing structural complexity of electronic components and systems makes fault diagnosis and testing a challenging task, particularly under the constraints of quality and cost. The motivation for test is to allow rapid diagnosis after a failure, thereby increasing the availability of a system (or of redundant system components). Design for test (DfT) and built-in self-test (BIST) facilities are increasingly being preferred over tricky ad-hoc design solutions to reduce test related difficulties. In addition to improved manufacturing test, BIST offers an extension towards on-field verification.

5.4.1 Principle of built-in self-test

According to a definition given in [33] "BIST is a design-for-test technique in which testing is accomplished through built-in hardware features". For chips with 1 million transistors and more the hardware required for testing can be integrated on the chip by dedicating a negligible percentage of the silicon (a quite optimistic estimation of 3% is given in [34]) for the BIST logic. Such test hardware is implemented in concurrence with the "actual" design. In this way BIST represents an important step towards regarding the test as one of the system functions.

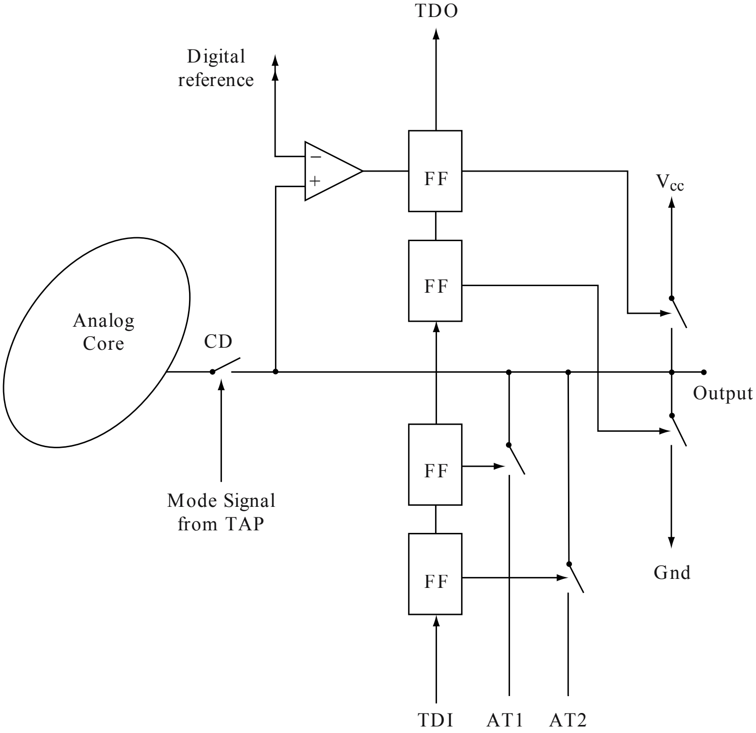


Figure 5.8: ***Details of an analog boundary c***

The principle of BIST is shown in Figure 5.9: A BIST Controller generates the test patterns, controls the CUT clock and collects and analyzes the responses. This makes the external test interface much more compact than in the conventional test approach: The self-test can be initiated over a single pin, the result ("accept" or "reject") can be signalled via a second pin. Optionally, a serial bit stream with diagnostic data may be provided at a third pin.

5.4.2 Advantages of BIST over conventional testing

The ATE required for the conventional factory test of VLSI circuits usually includes highly specific test hardware and probing solutions. Most often this expensive equipment can only be used for one specific product, and a reuse for higher level tests or during other test phases than factory test is not possible. As opposed to that, BIST-logic designed for a specific VLSI circuit can be extremely helpful for other test purposes like maintenance and diagnosis or start-up test. The combined utilization of hardware resources for concurrent checking and for off-line BIST is suggested in [35][36]. Moreover, BIST is ideally suited for hierarchical test structures (see Section 5.4.3, BIST and hierarchical testing).

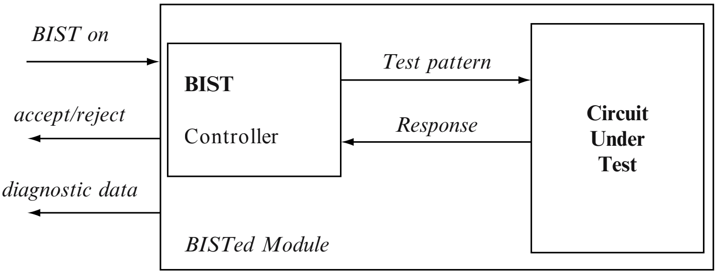


Figure 5.9: Principle of Built-in Self-test.

In the face of continuously shrinking package size and pitch, external probing becomes a virtually insoluble mechanical problem. At the same time system clock rates continue to rise and timing margins become narrower, therefore a test should also cover the dynamic properties of a circuit. Ideally, the test is performed at the nominal system clock rate (at-speed testing [37][38]), which also helps to keep test duration low. With a conventional external test approach, however, non-idealities in contacts and wiring deteriorate signal quality substantially, which prohibits at-speed testing. Moreover, the technology of the test equipment (clock rate, propagation delays) may be inferior to that of the CUT. These problems do not apply for BIST, since technology of CUT and tester are identical, probing is reduced to a minimum and timing of the interface signals is not critical. In some cases, however, the inclusion of the BIST logic exhibits a negative effect on system timing, resulting in the need for a minor reduction of the nominal system clock.

Obviously BIST improves testability of a design significantly: All nodes in the circuit are directly accessible and test points can be freely chosen as suggested by the test strategy without having to connect them to external pins. BIST facilitates a test-per-clock strategy, while conventional testing is often restricted to test-per-scan.

Another crucial cost factor is the degree of automation: VLSI design tools automatically generate the set of test vectors, but the adaptation or design process for the tester hardware and probing remains an expensive requirement for the conventional testing approach. As opposed to that there are suites of tools available that automatically add a complete and largely optimized BIST environment to a VHDL or Verilog-design [38].

Table 5.I summarizes the above discussion on the benefits of BIST. It is interesting to observe that BIST naturally unifies circuit design and test – two areas that have traditionally been largely disjoint. At the same time BIST moves test cost from equipment to design: The costly test equipment is traded for design overheads.

Table 5-I: Advantages of built-in self-test over conventional testing.

	Conventional Test	**Built-in self-test**
Probing	difficult	simple
Testability	very low	high
Cost of test equipment	high	few % area overhead
Technology	may be critical	identical to CUT
Test speed	problematic	“at-speed testing”
Test time	often very high	low
Total test cost	high	moderate
Applicability	specific	reusable
Automatic generation	stimuli only	complete

5.4.3 BIST and hierarchical testing

Perhaps the most significant benefit of BIST lies in its reusability on higher levels. A system level BIST may activate board level BIST on all system boards that, in turn, activate chip level BIST. The individual BIST results are combined and propagated through the hierarchy to finally form the system BIST result [33].

This approach has numerous advantages over the common software-implemented system test:

1. BIST localizes the testing problem: There is no need to carry the test stimulus through many layers and then convey the result back again.
2. As a design for testability method the expert who has the most detailed knowledge on the functionality of the design implements BIST already during circuit/board/system design. Triggering the

BIST from a higher level does not require any expertise on the lower level function.

3. BIST naturally supports the hierarchical test approach that is often applied in conventional tests to reduce complexity.
4. BIST allows for optimal diagnostic resolution.

The extra cost for system level BIST can be kept at a minimum, since a reuse of the factory BIST logic on chip level is possible [38].

5.5 Summary

Design-for-test (DfT) is the major area of interest among the test engineers of IC manufacturing semiconductor industry. In this chapter the authors have tried to discuss briefly all the major areas of DfT of analog circuits. The rules and guidelines were introduced in sub-section 5.2.1. The support of external test and the access to embedded blocks were discussed in the two subsequent sub-sections. On-chip test is also discussed in this chapter. The effect of test bus has been discussed. It follows by some standard test bus. The chapter is concluded with the Built-in self-test (BIST) system and its advantages.

References

[1] M. F. Toner and G. W. Roberts, "Histogram-Based Testing of a Sigma-Delta ADC", Proceedings, 32nd Midwest Symposium on Circuits and Systems, Washington, pp. 760-763, Aug. 1992.

[2] S.D. Millman, "Improving quality: yield versus test coverage", Journal of Electronic Testing: Theory and Applications, pp. 253-261, 1994.

[3] C. Mauder and R. Tulloss, The Test Access Port and Boundary Scan Architecture, IEEE Computer Society Press Tutorial, 1990.

[4] A. Grochowski, D. Bhattacharya, T.R. Viswanathan and K. Laker, "Integrated Circuit Testing for Quality Assurance in Manufacturing: History, Current Status and Future Trends", IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing, Vol. 44, pp. 610-633, August 1997.

[5] K. Baker, A.M. Richardson and A.P. Dorey, "Mixed signal test - techniques, applications and demands", IEE Proceedings - G, Circuits, Devices and Systems, Vol. 143, pp. 358-365, Dec. 1996.

[6] F.G.M. de Jong, "Analogue system testability: an industrial perspective", Proceedings, AMATIST ESPRIT Open Workshop, Twente, May 1997.

[7] S. Sunter, "Mixed-Signal BIST – Does Industry Need it?", Proceedings, 3rd IEEE International Mixed Signal Workshop, Tutorial 2, Seattle, June 1997.

[8] Semiconductor Industry Association (SIA). The National Technology Roadmap for Semiconductors, 1997.

[9] M.H. Costin, "On-board diagnostics of vehicle emission system components: review of upcoming government regulation", Proceedings, IFAC Fault Detection, Supervision and Safety for Technical Processes, Baden-Baden, pp. 497-501, 1991.

[10] M.Abramovici, M.A.Breuer, and A.D.Friedman, Digital System Testing and Testable Design, New York: Computer Science Press, 1990.

[11] K. D. Wagner and T. W. Williams, "Design for testability of analog/digital networks", IEEE Transactions on Industrial. Electronics, Vol. 36, pp. 227-230, May 1989.

[12] J.A Prieto, A Rueda, I Grout, E Peralias, J L Huertas and A M Richardson, "An Approach to Realistic Fault prediction and Layout Design for Testability in Analogue Circuits" Proceedings, Design, Automation & Test in Europe Conference, Paris, pp. 905-912, Feb. 1998.

[13] B. Atzema and T. Zwemstra, "Exploit Analog IFA to improve Specification Based Tests" Proceedings, European Design and Test Conference, Paris, paper 10C-1, March 1996.

[14] A. Lechner, A. Richardson, B. Hermes and M. Ohletz "A Design for Testability Study on a High Performance Automatic Gain Control Circuit", Proceedings, 16th IEEE VLSI Test Symposium (VTS98), Monterey, California, pp. 376-385, April 1998.

[15] J.M. Soden, C.F. Hawkins, R.K. Gulati, and W. Mao, "IDDQ Testing: A Review," Journal of Electronic Testing: Theory and Applications, Vol. 3, pp. 5-17, 1992.

[16] A.H. Bratt, R.J. Harvey, A.P. Dorey and A.M. Richardson "Aspects of Current Reference Generation and Distribution for IDDx Pass/Fail Current Determination" Proceedings, IEE Colloquium on Mixed Signal VLSI Test, London, Digest no 1993/240, pp. 3/1-3/8, Dec 1993.

[17] Andrew Richardson, Adrian Bratt, Iluminada Baturone and Jose Luis Huertas "The Application of IDDX Test Strategies in Analogue and Mixed Signal IC's", Proceedings, IEEE Mixed Signal Test Workshop, Grenoble, pp.206-211, June 1995.

[18] J.P.M van Lammeren, "ICCQ: A Test Method for Analogue VLSI Based on Current Monitoring", Proceedings, IEEE International Workshop on IDDQ Testing, Washington DC, pp. 24-29, November 1997.

[19] P.S.A. Evans, M.A. Al-Qutayri, and P.R. Sheperd, "A novel technique for testing mixed-signal ICs", Proceedings, European Test Conference, Munich, Germany, pp. 301-306, April 1991.

[20] G. Russell and D.S. Learmonth, "Systematic approaches to testing embedded analogue circuit functions", Microelectronics Journal, Vol. 25, pp. 133-138, March 1994.

[21] IEEE P1149.4 proposal "Standard for Mixed Signal Test Bus" March 1995, IEEE Standards Department, 445 Hoes Lane, PO Box 1331, Piscataway, NJ 08855-1331, USA.

[22] J.S. Matos, A.C. Leao, and J.C. Ferreira, "Control and observation of analog nodes in mixed-signal boards", Proceedings, International Test Conf., Baltimore, pp. 323-331, October 1993.

[23] L. Wurtz, "Built-in self-test structure for mixed-mode circuit", IEEE Transactions on Instrumentation and Measurement, Vol. 42, pp. 25-29, Feb. 1993.

[24] A. H. Bratt, A. M. D. Richardson, R. J. A. Harvey, and P. Dorey: "A design-for-test structure for optimising analogue and mixed signal IC test", Proceedings, European Design and Test Conference, Paris, pp. 24-32, March 1995.

[25] D. Vazquez, A. Rueda and J.L. Huertas, "A DfT methodology for active anaolgue filters", Proceedings, IEEE Mixed Signal Test Workshop, Grenoble, June 1995.

[26] D. Vázquez, A. Rueda, J. L. Huertas and E. Peralías. "A high-Q bandpass fully differential SC filter with enhanced testability", Proceedings, European Solid State Circuit Conference (ESSCIRC'97), Southampton, pp. 260-263, Sep. 1997.

[27] V. Kolarik, M. Lubaszewski, and B. Courtois, "Designing self-exercising analogue checkers", Proceedings, VLSI Test Symposium, 1994, pp. 252-257.

[28] A. Lechner, A. Richardson, A. Perkins, M. Zwolinski and B. Hermes, “Design for Testability Strategies for a High Performance Automatic Gain Control Circuit”, Proceedings, 4th IEEE International Mixed Signal Testing Workshop (IMSTW98), The Hague, The Netherlands, June 1998.

[29] K. Parker, The Boundary-Scan Handbook, Kluwer Academic Publishers, Massachusetts, 1992.

[30] R. Mason, B. Simon and K. Runtz, “Integrated Circuit Signal Measurements using an undersampling approach,” IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing, Vol. 45, pp. 1502-1504, Nov. 1998.

[31] S. Sunter, “A low cost 100 MHz analog test bus,” Proceedings, IEEE VLSI Test Symposium, pp. 60-63, 1995.

[32] K. Lofstrom, “A demonstration IC for the P1149.4 mixed-signal test standard,” Proceedings, IEEE International Test Conference, pp. 92-98, Oct. 1996.

[33] V. Agrawal, C. Kime and K. Saluja, “A Tutorial on Built-In Self-Test”, IEEE Design and Test of Computers, pp. 73-80, March 1993 and pp. 69-77, June 1993.

[34] P. Bardell, W. McAnney and J. Savir, Built-in Test for VLSI, Pseudorandom Techniques. John Wiley & Sons, New York, 1987.

[35] S. Gupta and, D. Pradhan, “Can concurrent Checkers help BIST?”, Proceedings, International Test Conference, Baltimore, pp. 140-150, Sept. 1992.

[36] B. Nadeau-Dostie, D. Burek and A. Hassan, “ScanBist: A Multifrequency Scan-Based BIST Method”, IEEE Design and Test of Computers, pp. 7-17, Spring 1994.

[37] M. Lubaszewski, B. Courtois, “On the Design of Self-Checking Boundary Scannable Boards”, Proceedings, International Test Conference, Baltimore, pp. 372-381, Sept. 1992.

[38] B. Könemann, B. Bennetts, N. Jarwala and B. Nadeau-Dostie, “Built-In Self-Test: Assuring System Integrity”, IEEE Computer, Vol. 29, pp. 39-45, November 1996.

Appendix A

Artificial Neural Networks: An Overview

A.1 Introduction

An Artificial Neural Network (ANN) is an interconnected network of a large number of elementary non-linear processors. These processors are referred to as neurons, because they resemble the biological neurons in two respects:

1. The network acquires knowledge through a learning process.
2. Inter-neuron connection strengths known as synaptic weights are used to store the knowledge.

The procedure used to perform the learning process is called a learning algorithm, the function of which is to modify the synaptic weights of the network in an orderly manner so as to attain a desired design objective [1] [2][3][4].

A.2 Model of an artificial neuron

Figure A.1 shows the schematic of the mathematical model of an artificial neuron. It consists of the following three basic elements:

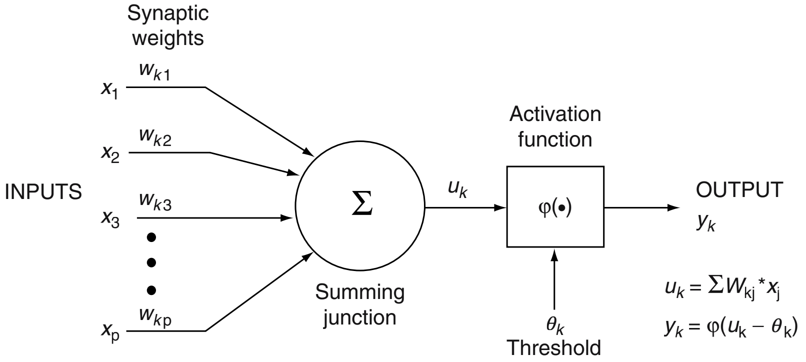


Figure A.1: Model of an artificial neuron.

1. A set of synapse or connecting links, each of which is characterised by a weight or strength of its own. Specifically, a signal, x_j, at the input of synapse, j, connected to a neuron, k, is multiplied by the synaptic weight, w_{kj}. The weight, w_{kj}, is positive if the associated synapse is excitatory and it is negative if the synapse is inhibitory.
2. An adder for summing the input signals weighted by the respective synapses of the neuron.

An activation function for limiting the amplitude of the output of a neuron. The activation function limits the permissible amplitude range of the output signal to some finite value. Typically, the normalised amplitude of the output of a neuron is written as the closed interval [0,1] or [-1,1].

The neuron model also includes an externally applied threshold, θ_k that has the effect of lowering the net input of the activation function. Employing a bias term rather than a threshold may increase the net input of the activation function. The bias is the negative of the threshold.

In mathematical terms, a neuron, k, may be described by writing the following pair of equations:

$$u_k = \sum w_{kj} * x_j \tag{A.1}$$

$$y_k = \varphi(u_k - \theta_k) \tag{A.2}$$

where $x_1, x_2, x_3, \ldots x_p$ are the input signals; $w_{k1}, w_{k2}, w_{k3}, \ldots w_{kp}$ are the synaptic weights of neuron, k; u_k is the output of the linear combiner; θ_k is the threshold; $\varphi(\bullet)$ is the activation function and y_k is the output signal of the neuron.

A.3 The activation function

The activation function denoted by $\varphi(\bullet)$, defines the output of a neuron in terms of the activity level at its input. This function typically falls into one of three categories:

- **linear** (or **ramp**)**:** For **linear units**, the output activity is proportional to the total weighted output.
- **threshold:** For **threshold units**, the output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.
- **sigmoid:** For **sigmoid units**, the output varies continuously but not linearly as the input changes. Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations.

Selected types of activation functions are shown in Figure A.2.

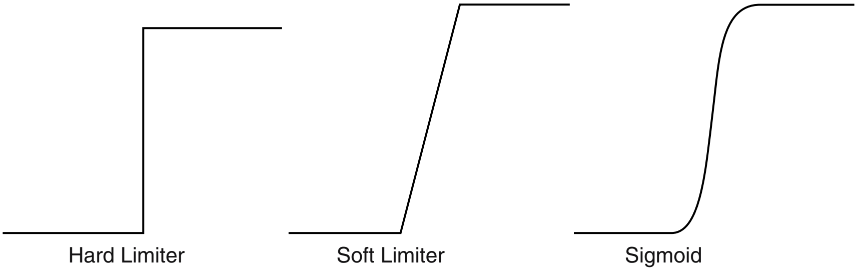


Figure A.2: Selected types of activation functions.

A.4 Structure of an artificial neural network

The structure of a neuron in an ANN is explained in Figure A.1. Information (called the input) is sent to the neuron on its incoming weights. This input is processed by a propagation function that adds up the values of all incoming weights. The resulting value is compared with a certain threshold value by the neuron's activation function. If the input exceeds the threshold value, the neuron will be activated, otherwise it will remain inhibited. If activated, the neuron sends an output on its outgoing weights to all connected neurons and so on.

In an ANN, the neurons are grouped in layers, called neuron layers. Usually each neuron of one layer is connected to all neurons of the preceding and the following layer (except the input layer and the output layer of the network). The information given to a neural network is propagated layer-by-layer from input layer to output layer either through none, one or more hidden layers. Depending on the learning algorithm it is also possible that information is propagated backwards through the network.

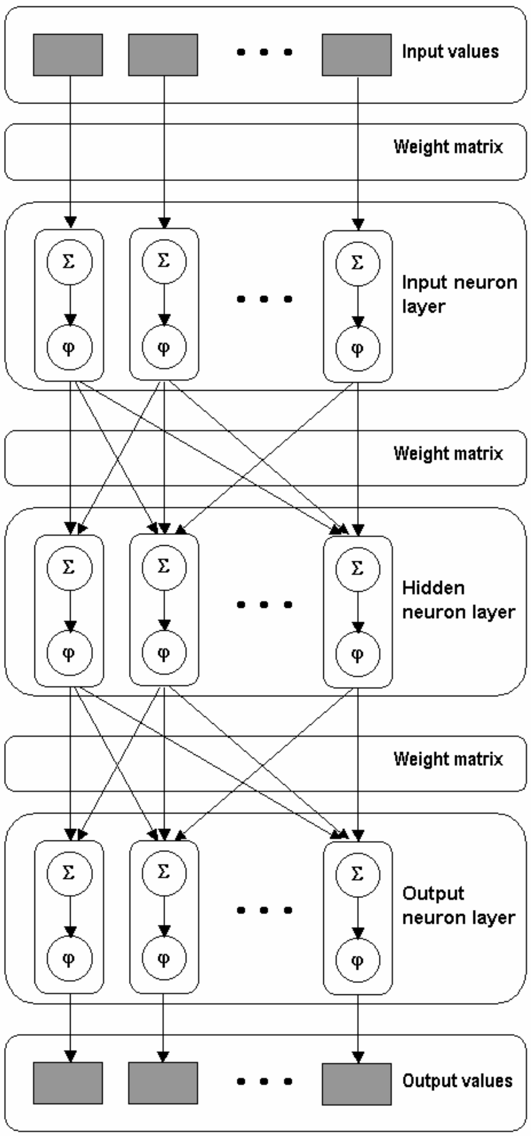


Figure A.3: ***An artificial neural network with three neuron layers.***

Figure A.3 shows an ANN with three neuron layers. There could be variations to the above. For example, some types of ANN have no hidden layers or the neurons in a layer are arranged as a matrix. What is common to all types of ANN is the presence of at least one weight matrix and the connections between two neuron layers. There are many different types of ANN with each having special properties, so each problem domain has its own type of network. They can be distinguished by

i. their type (feed-forward or feedback),
ii. their structure and
iii. the learning algorithm they use.

The *type* of a neural network indicates how the neurons of one of the network's layers may be connected among each other. Feed-forward neural networks allow only neuron connections between two different layers, while networks of the feedback type have also connections between neurons of the same layer.

In the scheme proposed in this book the simple feed-forward multi-layer ANN architecture trained with back-propagation algorithm is used to model the faulty and fault-free analog circuits.

References

[1] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, 2nd edition, 1998.

[2] Christopher M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.

[3] Kevin Swingler, Applying Neural Networks: A Practical Guide, Morgan Kaufmann, Book and Disk edition, 1996.

[4] H. Demuth and M. Beale, Neural Network Toolbox User's Guide, The MathWorks, Inc., Natick, 1994.

Appendix B

Typical BJT SPICE Parameter Values of the μA 741 Op-Amp

Table B.1: ***Model parameters of small and large n-p-n BJT [1][2].***

Parameters	**Description**	**Small n-p-n**	**Large n-p-n**
IS (A)	transport saturation current	1.26e-15	3.95e-16
BF	ideal maximum forward beta	209.0	400.0
NF (default) (typical)	forward current emission coefficient	1.0 1.0	1.0 1.0
VAF (V)	forward Early voltage	178.6	267.0
IKF (A)	forward-beta high-current roll-off “knee” current	1.611e-3	0.01
ISE (A)	base-emitter leakage saturation current	1.653e3	1.543e3
NE (default) (typical)	base-emitter leakage emission coefficient	1.5 2.0	1.5 2.0
BR	ideal maximum reverse beta	2.5	6.1
NR (default)	reverse current emission coefficient	1.0	1.0

Parameters	**Description**	**Small n-p-n**	**Large n-p-n**
VAR (default) (typical) (V)	reverse Early voltage	∞ 100	∞ 100
IKR (default) (typical) (A)	corner for reverse-beta high-current roll-off	∞ 0.1	∞ 0.1
ISC (default) (typical) (A)	base-collector leakage saturation current	0 1.0	0 1.0
NC (default) (typical)	base-collector leakage emission coefficient	2.0 2.0	2.0 2.0
NK (default) (typical)	high-current roll-off coefficient	0.5 0.5	0.5 0.5
RB (Ω)	zero-bias (maximum) base resistance	670.0	185.0
RBM (default) (typical) (Ω)	maximum base resistance	RB 100.0	RB 100.0
RE (default) (typical) (Ω)	emitter ohmic resistance	0 1.0	0 1.0
RC (Ω)	collector ohmic resistance	300.0	15.0
CJE (pF)	base-emitter zero-bias p-n capacitance	0.65	2.8
VJE (V)	base-emitter built-in potential	0.60	0.60
MJE	base-emitter p-n grading factor	3.0	3.0
CJC (pF)	base-collector zero-bias p-n capacitance	0.36	1.55
VJC (V)	base-collector built-in potential	0.45	0.45
MJC	base-collector p-n grading factor	3.0	3.0
XCJC (default)	fraction of C_{BC} connected internal to R_B	1.0	1.0
CJS (pF)	substrate zero-bias p-n capacitance	3.2	7.8
VJS (V)	substrate built-in potential	0.6	0.6
MJS	substrate p-n grading factor	4.0	4.0
FC (default)	forward-bias depletion capacitor coefficient	0.5	0.5
TF (sec)	ideal forward transit time	1.15e-9	7.6e-10
XTF (default)	transit time bias dependence coefficient	0	0

Parameters	Description	Small n-p-n	Large n-p-n
VTF (default) (typical) (V)	transit time dependency on V_{BC}	∞ 1.0e5	∞ 1.0e5
ITF (default) (A)	transit time dependency on I_C	0	0
PTF (default) (typical) (°)	excess phase at 1/(2π×TF) Hz	0 30	0 30
TR (sec)	ideal reverse transit time	4.05e-7	2.43e-7
EG (default) (typical) (eV)	band gap voltage (barrier height)	1.11 1.11	1.11 1.11
XTB (default)	forward and reverse beta temperature coefficient	0	0
XTI (default)	IS temperature effect exponent	3.0	3.0
KF (default) (typical)	flicker noise coefficient	0 6.6e-16	0 6.6e-16
AF (default) (typical)	flicker noise exponent	1.0 1.0	1.0 1.0

Table B.2: ***Model parameters of lateral and substrate p-n-p BJT [1][2].***

Parameters	Description	Lateral p-n-p	Substrate p-n-p
IS (A)	transport saturation current	3.15e-15	1.76e-14
BF	ideal maximum forward beta	75.0	117.0
NF (default) (typical)	forward current emission coefficient	1.0 1.0	1.0 1.0
VAF (V)	forward Early voltage	55.11	57.94
IKF (A)	forward-beta high-current roll-off “knee” current	2.70e-4	5.907e-4
ISE (A)	base-emitter leakage saturation current	1.764e3	478.4
NE (default) (typical)	base-emitter leakage emission coefficient	1.5 2.0	1.5 2.0
BR	ideal maximum reverse beta	3.8	4.8
NR (default)	reverse current emission coefficient	1.0	1.0
VAR (default) (typical) (V)	reverse Early voltage	∞ 100	∞ 100

Parameters	**Description**	**Lateral p-n-p**	**Substrate p-n-p**
IKR (default) (typical) (A)	corner for reverse-beta high-current roll-off	∞ 0.1	∞ 0.1
ISC (default) (typical) (A)	base-collector leakage saturation current	0 1.0	0 1.0
NC (default) (typical)	base-collector leakage emission coefficient	2.0 2.0	2.0 2.0
NK (default) (typical)	high-current roll-off coefficient	0.5 0.5	0.5 0.5
RB (Ω)	zero-bias (maximum) base resistance	500.0	80.0
RBM (default) (typical) (Ω)	maximum base resistance	RB 100.0	RB 100.0
RE (default) (typical) (Ω)	emitter ohmic resistance	0 1.0	0 1.0
RC (Ω)	collector ohmic resistance	150.0	156.0
CJE (pF)	base-emitter zero-bias p-n capacitance	0.10	4.05
VJE (V)	base-emitter built-in potential	0.45	0.60
MJE	base-emitter p-n grading factor	3.0	4.0
CJC (pF)	base-collector zero-bias p-n capacitance	1.05	2.8
VJC (V)	base-collector built-in potential	0.45	0.60
MJC	base-collector p-n grading factor	3.0	4.0
XCJC (default)	fraction of C_{BC} connected internal to R_B	1.0	1.0
CJS (pF)	substrate zero-bias p-n capacitance	5.1	0 (default)
VJS (V)	substrate built-in potential	0.6	0.75 (default)
MJS	substrate p-n grading factor	4.0	0 (default)
FC (default)	forward-bias depletion capacitor coefficient	0.5	0.5
TF (sec)	ideal forward transit time	2.74e-8	2.65e-8
XTF (default)	transit time bias dependence coefficient	0	0
VTF (default) (typical) (V)	transit time dependency on V_{BC}	∞ 1.0e5	∞ 1.0e5

Parameters	**Description**	**Lateral p-n-p**	**Substrate p-n-p**
ITF (default) (A)	transit time dependency on I_C	0	0
PTF (default) (typical) (°)	excess phase at $1/(2\pi\times TF)$ Hz	0 30	0 30
TR (sec)	ideal reverse transit time	2.54e-6	2.43e-6
EG (default) (typical) (eV)	band gap voltage (barrier height)	1.11 1.11	1.11 1.11
XTB (default)	forward and reverse beta temperature coefficient	0	0
XTI (default)	IS temperature effect exponent	3.0	3.0
KF (default) (typical)	flicker noise coefficient	0 6.6e-16	0 6.6e-16
AF (default) (typical)	flicker noise exponent	1.0 1.0	1.0 1.0

Table B.3: ***Model Parameters of double collector lateral p-n-p BJT [1][2].***

Parameters	**Description**	**Double Collector Lateral p-n-p BJT**	
		A	**B**
IS (A)	transport saturation current	9.0e-16	2.25e-15
BF	ideal maximum forward beta	0.38	1.48
NF (default) (typical)	forward current emission coefficient	1.0 1.0	1.0 1.0
VAF (V)	forward Early voltage	83.55	83.55
IKF (A)	forward-beta high-current roll-off “knee” current	5.0e-3	1.718e-4
ISE (A)	base-emitter leakage saturation current	8.437e4	8.437e4
NE (default) (typical)	base-emitter leakage emission coefficient	1.5 2.0	1.5 2.0
BR	ideal maximum reverse beta	1.4	1.5
NR (default)	reverse current emission coefficient	1.0	1.0
VAR (default) (typical) (V)	reverse Early voltage	∞ 100	∞ 100

Parameters	**Description**	**Double Collector Lateral p-n-p BJT**	
		A	**B**
IKR (default) (typical) (A)	corner for reverse-beta high-current roll-off	∞ 0.1	∞ 0.1
ISC (default) (typical) (A)	base-collector leakage saturation current	0 1.0	0 1.0
NC (default) (typical)	base-collector leakage emission coefficient	2.0 2.0	2.0 2.0
NK (default) (typical)	high-current roll-off coefficient	0.5 0.5	0.5 0.5
RB (Ω)	zero-bias (maximum) base resistance	1.0e3	1.6e3
RBM (default) (typical) (Ω)	maximum base resistance	RB 100.0	RB 100.0
RE (default) (typical) (Ω)	emitter ohmic resistance	0 1.0	0 1.0
RC (Ω)	collector ohmic resistance	80.0	120.0
CJE (pF)	base-emitter zero-bias p-n capacitance	0.10	0.10
VJE (V)	base-emitter built-in potential	0.45	0.45
MJE	base-emitter p-n grading factor	3.0	3.0
CJC (pF)	base-collector zero-bias p-n capacitance	0.3	0.9
VJC (V)	base-collector built-in potential	0.45	0.45
MJC	base-collector p-n grading factor	3.0	3.0
XCJC (default)	fraction of C_{BC} connected internal to R_B	1.0	1.0
CJS (pF)	substrate zero-bias p-n capacitance	4.8	4.8
VJS (V)	substrate built-in potential	0.6	0.6
MJS	substrate p-n grading factor	4.0	4.0
FC (default)	forward-bias depletion capacitor coefficient	0.5	0.5
TF (sec)	ideal forward transit time	2.74e-8	2.74e-8
XTF (default)	transit time bias dependence coefficient	0	0
VTF (default) (typical) (V)	transit time dependency on V_{BC}	∞ 1.0e5	∞ 1.0e5
ITF (default) (A)	transit time dependency on I_C	0	0

Parameters	Description	Double Collector Lateral p-n-p BJT	
		A	**B**
PTF (default) (typical) (°)	excess phase at 1/(2π×TF) Hz	0 30	0 30
TR (sec)	ideal reverse transit time	5.5e-8	2.2e-7
EG(default) (typical)(eV)	band gap voltage (barrier height)	1.11 1.11	1.11 1.11
XTB(default)	forward and reverse beta temperature coefficient	0	0
XTI(default)	IS temperature effect exponent	3.0	3.0
KF(default) (typical)	flicker noise coefficient	0 6.6e-16	0 6.6e-16
AF(default) (typical)	flicker noise exponent	1.0 1.0	1.0 1.0

Table B.4: Model parameters of double emitter substrate p-n-p BJT [1][2].

Parameters	Description	Double Emitter Substrate p-n-p BJT	
		A	**B**
IS (A)	transport saturation current	7.9e-16	6.3e-18
BF	ideal maximum forward beta	80	19
NF (default) (typical)	forward current emission coefficient	1.0 1.0	1.0 1.0
VAF (V)	forward Early voltage	79.45	167.1
IKF (A)	forward-beta high-current roll-off “knee” current	8.055e-5	8.055e-5
ISE (A)	base-emitter leakage saturation current	1.219e3	5.749e4
NE (default) (typical)	base-emitter leakage emission coefficient	1.5 2.0	1.5 2.0
BR	ideal maximum reverse beta	1.5	1.0
NR (default)	reverse current emission coefficient	1.0	1.0
VAR (default) (typical) (V)	reverse Early voltage	∞ 100	∞ 100
IKR (default) (typical) (A)	corner for reverse-beta high-current roll-off	∞ 0.1	∞ 0.1

Parameters	**Description**	**Double Emitter Substrate p-n-p BJT**	
		A	**B**
ISC (default) (typical) (A)	base-collector leakage saturation current	0 1.0	0 1.0
NC (default) (typical)	base-collector leakage emission coefficient	2.0 2.0	2.0 2.0
NK (default) (typical)	high-current roll-off coefficient	0.5 0.5	0.5 0.5
RB (Ω)	zero-bias (maximum) base resistance	1.1e3	650
RBM (default) (typical) (Ω)	maximum base resistance	RB 100.0	RB 100.0
RE (default) (typical) (Ω)	emitter ohmic resistance	0 1.0	0 1.0
RC (Ω)	collector ohmic resistance	170.0	100.0
CJE (pF)	base-emitter zero-bias p-n capacitance	1.1	1.9
VJE (V)	base-emitter built-in potential	0.6	0.6
MJE	base-emitter p-n grading factor	4.0	4.0
CJC (pF)	base-collector zero-bias p-n capacitance	2.4	2.4
VJC (V)	base-collector built-in potential	0.6	0.6
MJC	base-collector p-n grading factor	4.0	4.0
XCJC (default)	fraction of C_{BC} connected internal to R_B	1.0	1.0
CJS (default) (pF)	substrate zero-bias p-n capacitance	0	0
VJS (default) (V)	substrate built-in potential	0.75	0.75
MJS (default)	substrate p-n grading factor	0	0
FC (default)	forward-bias depletion capacitor coefficient	0.5	0.5
TF (sec)	ideal forward transit time	2.65e-8	2.65e-8
XTF (default)	transit time bias dependence coefficient	0	0
VTF (default) (typical) (V)	transit time dependency on V_{BC}	∞ 1.0e5	∞ 1.0e5
ITF (default) (A)	transit time dependency on I_C	0	0

Parameters	Description	Double Emitter Substrate p-n-p BJT	
		A	B
PTF (default) (typical) (°)	excess phase at 1/(2π×TF) Hz	0 30	0 30
TR (sec)	ideal reverse transit time	9.55e-6	2.12e-6
EG(default) (typical)(eV)	band gap voltage (barrier height)	1.11 1.11	1.11 1.11
XTB (default)	forward and reverse beta temperature coefficient	0	0
XTI (default)	IS temperature effect exponent	3.0	3.0
KF (default) (typical)	flicker noise coefficient	0 6.6e-16	0 6.6e-16
AF (default) (typical)	flicker noise exponent	1.0 1.0	1.0 1.0

References

[1] B.A. Wooley, S J. Wong and D O. Pederson, "A computer-aided evaluation of the 741 amplifier", *IEEE Journal of Solid-State Circuits*, Vol. SC-6, No. 6, pp. 357-366, December 1971.

[2] P.W. Tuinenga, SPICE: A Guide to Circuit Simulation and Analysis using PSpice, Prentice-Hall, New Jersey, 1992.

Appendix C

Typical MOSFET SPICE Parameter Values of the MOS Op-Amp

Table C.1: ***Model parameters of p-channel MOSFET PMOS1 and PMOS2 [1][2].***

Parameters	**Description**	**PMOS1**	**PMOS2**
LEVEL	model type (1, 2, or 3)	3	3
L (µmeter)	channel length	4.0	4.0
W (µmeter)	channel width	150.0	35.0
LD (meter)	lateral diffusion length	6.0e-8	6.0e-8
WD (default) (typical) (meter)	lateral diffusion width	0 0	0 0
VTO (V)	zero-bias threshold voltage	-0.84	-0.84
KP (default) (typical) (A/V^2)	transconductance	2.0e-5 2.5e-5	2.0e-5 2.5e-5
GAMMA ($V^{1/2}$)	bulk threshold parameter	0.53	0.53
PHI (V)	surface potential	0.58	0.58

Parameters	**Description**	**PMOS1**	**PMOS2**
RD (Ω)	drain ohmic resistance	94	94
RS (Ω)	source ohmic resistance	94	94
RG (default) (typical) (Ω)	gate ohmic resistance	0 1	0 1
RB (default) (typical) (Ω)	bulk ohmic resistance	0 1	0 1
RDS (default) (typical) (Ω)	drain-source shunt resistance	∞ 1.0e6	∞ 1.0e6
RSH (Ω/square)	drain-source diffusion sheet resistance	100	100
IS (A)	bulk p-n saturation current	1.0e-16	1.0e-16
JS (A/meter^2)	bulk p-n saturation current/ area	1.0e-4	1.0e-4
PB (V)	bulk p-n potential	0.8	0.8
CBD (default) (typical) (pF)	bulk-drain zero-bias p-n capacitance	0 5.0	0 5.0
CBS (default) (typical) (pF)	bulk-source zero-bias p-n capacitance	0 2.0	0 2.0
CJ (F/meter^2)	bulk p-n zero-bias bottom capacitance per unit area	4.1e-4	4.1e-4
CJSW (pF/meter)	bulk p-n zero-bias sidewall capacitance per unit length	340	340
MJ	bulk p-n bottom grading coefficient	0.54	0.54
MJSW	bulk p-n sidewall grading coefficient	0.3	0.3
FC	bulk p-n forward-bias capacitance coefficient	0.5	0.5
CGSO (pF/meter^2)	gate-source overlap capacitance/ channel width	328.4	328.4
CGDO (pF/meter^2)	gate-drain overlap capacitance/ channel width	328.4	328.4
CGBO (pF/meter^2)	gate-bulk overlap capacitance/ channel width	0.5	0.5
NSUB (1/cm^3)	substrate doping density	1.75e16	1.75e16
NFS (1/cm^2)	fast surface state density	8.4e11	8.4e11
TOX (nano-meter)	oxide thickness	25	25

Parameters	**Description**	**PMOS1**	**PMOS2**
TPG	gate material type	+1	+1
XJ (meter)	metallurgical junction depth	0	0
UO (cm^2/V-sec)	surface mobility	205	205
VMAX (meter/sec)	maximum drift velocity	5.0e5	5.0e5
DELTA	width effect on threshold	0.4598	0.4598
THETA (V^{-1})	mobility modulation	0.14	0.14
ETA	static feedback	0.17	0.17
KAPPA	saturation field factor	10	10
KF (default) (typical)	flicker noise coefficient	0 1.0e-26	0 1.0e-26
AF (default) (typical)	flicker noise exponent	1 1.2	1 1.2

Table C.2: Model parameters of p-channel MOSFET PMOS3 and PMOS4 [1][2].

Parameters	**Description**	**PMOS3**	**PMOS4**
LEVEL	model type (1, 2, or 3)	3	3
L (μmeter)	channel length	4.0	4.0
W (μmeter)	channel width	100.0	60.0
LD (meter)	lateral diffusion length	6.0e-8	6.0e-8
WD (default) (typical) (meter)	lateral diffusion width	0 0	0 0
VTO (V)	zero-bias threshold voltage	-0.84	-0.84
KP (default) (typical) (A/V^2)	transconductance	2.0e-5 2.5e-5	2.0e-5 2.5e-5
GAMMA (V$^{1/2}$)	bulk threshold parameter	0.53	0.53
PHI (V)	surface potential	0.58	0.58
RD (Ω)	drain ohmic resistance	94	94
RS (Ω)	source ohmic resistance	94	94
RG (default) (typical) (Ω)	gate ohmic resistance	0 1	0 1
RB (default) (typical) (Ω)	bulk ohmic resistance	0 1	0 1

Parameters	**Description**	**PMOS3**	**PMOS4**
RDS (default) (typical) (Ω)	drain-source shunt resistance	∞ 1.0e6	∞ 1.0e6
RSH (Ω/square)	drain-source diffusion sheet resistance	100	100
IS (A)	bulk p-n saturation current	1.0e-16	1.0e-16
JS (A/meter^2)	bulk p-n saturation current/ area	1.0e-4	1.0e-4
PB (V)	bulk p-n potential	0.8	0.8
CBD (default) (typical) (pF)	bulk-drain zero-bias p-n capacitance	0 5.0	0 5.0
CBS (default) (typical) (pF)	bulk-source zero-bias p-n capacitance	0 2.0	0 2.0
CJ (F/meter^2)	bulk p-n zero-bias bottom capacitance per unit area	4.1e-4	4.1e-4
CJSW (pF/meter)	bulk p-n zero-bias sidewall capacitance per unit length	340	340
MJ	bulk p-n bottom grading coefficient	0.54	0.54
MJSW	bulk p-n sidewall grading coefficient	0.3	0.3
FC	bulk p-n forward-bias capacitance coefficient	0.5	0.5
CGSO (pF/meter^2)	gate-source overlap capacitance/ channel width	328.4	328.4
CGDO (pF/meter^2)	gate-drain overlap capacitance/ channel width	328.4	328.4
CGBO (pF/meter^2)	gate-bulk overlap capacitance/ channel width	0.5	0.5
NSUB (1/cm^3)	substrate doping density	1.75e16	1.75e16
NFS (1/cm^2)	fast surface state density	8.4e11	8.4e11
TOX (nano-meter)	oxide thickness	25	25
TPG	gate material type	+1	+1
XJ (meter)	metallurgical junction depth	0	0
UO (cm^2/V-sec)	surface mobility	205	205
VMAX (meter/sec)	maximum drift velocity	5.0e5	5.0e5

Parameters	Description	PMOS3	PMOS4
DELTA	width effect on threshold	0.4598	0.4598
THETA (V^{-1})	mobility modulation	0.14	0.14
ETA	static feedback	0.17	0.17
KAPPA	saturation field factor	10	10
KF (default) (typical)	flicker noise coefficient	0 1.0e-26	0 1.0e-26
AF (default) (typical)	flicker noise exponent	1 1.2	1 1.2

Table C.3: ***Model parameters of n-channel MOSFET NMOS1 and NMOS2 [1][2].***

Parameters	Description	NMOS1	NMOS2
LEVEL	model type (1, 2, or 3)	3	3
L (μmeter)	channel length	4.0	4.0
W (μmeter)	channel width	27.5	100.0
LD (meter)	lateral diffusion length	7.0e-8	7.0e-8
WD (default) (typical) (meter)	lateral diffusion width	0 0	0 0
VTO (V)	zero-bias threshold voltage	0.79	0.79
KP (default) (typical) (A/V^2)	transconductance	2.0e-5 2.5e-5	2.0e-5 2.5e-5
GAMMA ($V^{1/2}$)	bulk threshold parameter	0.38	0.38
PHI (V)	surface potential	0.53	0.53
RD (Ω)	drain ohmic resistance	63	63
RS (Ω)	source ohmic resistance	63	63
RG (default) (typical) (Ω)	gate ohmic resistance	0 1	0 1
RB (default) (typical) (Ω)	bulk ohmic resistance	0 1	0 1
RDS (default) (typical) (Ω)	drain-source shunt resistance	∞ 1.0e6	∞ 1.0e6
RSH (Ω/square)	drain-source diffusion sheet resistance	45	45
IS (A)	bulk p-n saturation current	1.0e-16	1.0e-16

Parameters	**Description**	**NMOS1**	**NMOS2**
JS (A/meter^2)	bulk p-n saturation current/ area	1.0e-4	1.0e-4
PB (V)	bulk p-n potential	0.8	0.8
CBD (default) (typical) (pF)	bulk-drain zero-bias p-n capacitance	0 5.0	0 5.0
CBS (default) (typical) (pF)	bulk-source zero-bias p-n capacitance	0 2.0	0 2.0
CJ (F/meter^2)	bulk p-n zero-bias bottom capacitance per unit area	2.9e-4	2.9e-4
CJSW (pF/meter)	bulk p-n zero-bias sidewall capacitance per unit length	330	330
MJ	bulk p-n bottom grading coefficient	0.486	0.486
MJSW	bulk p-n sidewall grading coefficient	0.33	0.33
FC	bulk p-n forward-bias capacitance coefficient	0.5	0.5
CGSO (pF/meter^2)	gate-source overlap capacitance/ channel width	197.3	197.3
CGDO (pF/meter^2)	gate-drain overlap capacitance/ channel width	197.3	197.3
CGBO (pF/meter^2)	gate-bulk overlap capacitance/ channel width	0.5	0.5
NSUB (1/cm^3)	substrate doping density	8.7e15	8.7e15
NFS (1/cm^2)	fast surface state density	8.2e11	8.2e11
TOX (nano-meter)	oxide thickness	25	25
TPG	gate material type	+1	+1
XJ (meter)	metallurgical junction depth	1.0e-7	1.0e-7
UO (cm^2/V-sec)	surface mobility	577	577
VMAX (meter/sec)	maximum drift velocity	1.5e5	1.5e5
DELTA	width effect on threshold	0.3551	0.3551
THETA (V^{-1})	mobility modulation	0.046	0.046
ETA	static feedback	0.16	0.16
KAPPA	saturation field factor	0.05	0.05

Parameters	**Description**	**NMOS1**	**NMOS2**
KF (default) (typical)	flicker noise coefficient	0 1.0e-26	0 1.0e-26
AF (default) (typical)	flicker noise exponent	1 1.2	1 1.2

References

[1] P.W. Tuinenga, SPICE: A Guide to Circuit Simulation and Analysis using PSpice, Prentice-Hall, New Jersey, 1992.

[2] IEEE mixed-signal benchmark circuit home page available on the web-page at: *http://faculty.washington.edu/mani soma/madtest/benchmarks/OpAmp.htm.*

Notes

Index