This is to certify that the

dissertation entitled

Fault Prediction and Diagnosis
In Large Analog Circuit Networks

presented by

Benlu Jiang

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in Electrical Engineering

Major professor

Date July 10, 1989

FAULT PREDICTION AND DIAGNOSIS IN LARGE ANALOG CIRCUIT NETWORKS

By

Benlu Jiang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1989

ABSTRACT

FAULT PREDICTION AND DIAGNOSIS IN LARGE ANALOG CIRCUIT NETWORKS

By

Benlu Jiang

Electronic circuits and systems have become so versatile and useful that they are indispensable in modern society. With the electronic systems continuously growing in the significance and pervasiveness of application, their testing becomes increasingly important and also more and more complex, difficult, and costly.

Most of the automatic testing and fault diagnosis algorithms proposed in literature work well only when nonfaulty components assume their nominal values exactly. In a practical circuit, the values of nonfault components may deviate from their nominal values within a predefined tolerance, and the faulty components may not be properly located or even be detected unless the system fails. This becomes the bottleneck of analog fault diagnosis. For a reliable design, we should identify potential faulty components as soon as possible before the system fails instead of doing fault location after the system fails.

In this dissertation, the fault prediction problem is initiated and a fault prediction algorithm is presented. For the case that the parameters of potential faulty components are assumed to change gradually during each maintenance period, by continuously monitoring the responses of the network, the proposed algorithm can precisely predict whether any of the network components are about to fail.

In order to apply the proposed algorithms to large circuit networks with reasonably high speed, a decomposition approach fault prediction algorithm is proposed. The approach can be used hierarchically to decompose a network into any desired level to predict and diagnose faulty subnetworks.

Due to technical limitation, it is difficult to provide proportionately more accessible terminals for testing purpose in large circuit networks. To deal with this problem, an analog build-in self-test (ABIST) structure is proposed which can provide more test points while still keeping low pin overhead and acquire test data at various test points simultaneously. It is the first analog BIST structure ever proposed for analog fault diagnosis. In order to properly design diagnosable networks, an efficient algorithm is developed to select an appropriate minimum set of test points.

In summary, this dissertation focuses on fault prediction and diagnosis. The proposed decomposition approach, ABIST structure and diagnosability design provide a useful means for fault prediction and diagnosis in large analog circuit networks.

To my parents:

Dazong Jiang and *Zongxin Huang*

ACKNOWLEDGEMENTS

I would like to thank Professor Chin-Long Wey, my thesis advisor, for his valuable inspiration and guidance throughout my graduate study. I would like to thank members of my thesis committee, Professors Greg Wierzba, Donald Reinhard, Edwin Kashy, and Michael Shanblatt, for their continual encouragement and many excellent comments during the course of my dissertation research.

I would like to acknowledge all the faculty members and students who gave me help and assistance during my studying at Michigan State University, and many friends who showed their support and concern.

I am very grateful to my parents Dazong Jiang and Zongxin Huang, for their years of concern, encouragement and support; to my husband Youran for his constant encouragement and direct assistance in the preparing of this manuscript; and to my daughter Lana for being so understanding.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Electric circuits and systems have become so versatile and useful that they are indispensable in modern society. With the electronic systems continually growing in the significance and pervasiveness of application, their testing becomes increasingly important and also more and more complex, difficult, and costly.

Electronics design has become more and more sophisticated since the last quarter of this century. Graphical algorithms have been replaced by CAD (Computer-Aided Design), and features of design implementation can be studied by simulation rather than requiring extensive breadboarding. Electronics maintenance, however, has changed very little during the same period. Therefore, our ability to design and manufacture complex electronic circuits is quickly outstripping our ability to maintain them, and the price reductions which have accompanied the new electronics technology are being offset by increased maintenance costs. Indeed, many industries are finding that the life cycle maintenance costs for their electronic equipment now exceeds their original capitol investment.

It is becoming apparent that the electronics maintenance process, like the design process, must be automated. For more than two decades, the subjects of automatic testing and fault diagnosis of electronic circuits have been of interest to researchers in the areas of circuits and systems [8, 25, 32, 33, 38]. Recently, with the rapidly increasing complexity and size of modern electronic systems, these subjects become more and

more important and critical.

The main problems in network testing are *fault detection, fault location* and *fault prediction* [3]. Fault detection refers to the discovery of something wrong in a circuit. Fault location concerns the identification of faults with components, functional modules, or subsystems, depending upon the requirement. Fault prediction alludes to the continuous monitoring of network responses so that any of the network elements which is about to fail can be identified. Fault diagnosis includes fault detection and location.

By a *fault* we mean, in general, any variation in component value, with respect to its nominal value which can cause the failure of the whole circuit. Two forms of faults are generally considered: *catastrophic fault (hard fault)*, where the faulty element produces either a short circuit or an open circuit, and *deviation fault (soft fault)*, where the faulty element deviates from its nominal value without reaching its extreme bounds. The soft faults are usually due to manufacturing tolerances, aging, or parasitic effects.

1.1 DIFFICULTY OF ANALOG FAULT DIAGNOSIS

The research and theory development of digital testing started in the mid 1960's when the large-scale computers were readily available. The first commercialized test program did not become available until a decade later. Several digital automatic test program generation systems have been developed and widely used by both the military and industrial communities. From the mid 1970's, the test technology community began to face up to the analog test problem. Indeed, even in a predominantly digital world, analog systems were not disappearing. Analog systems were proving to be among the most unreliable and least readily tested of all electronic systems [26]. Presently, it is estimated that even though 80% of all boards are digital, 80% of the problems are analog.

Given our experience with the digital test problem and the analog computer-aided design problem, one might initially assume that the analog test problem could be resolved simply by integrating the tools and techniques of these two well-established fields. Unfortunately, the tremendous strides which have been made in digital test technology have not been paralleled by equal progress in the analog area. Fault diagnosis for analog circuits, in fact, has been found to be an extremely difficult problem to solve. The difficulty arises from a number of characteristics of the analog problem [3, 26] as listed below:

(1) *Tolerance*: The actual values of analog components almost always deviate from the nominal values.

(2) *Modeling*: An analog system has an infinite number of possible failures which may range from short circuit to open circuit. It leads the lack of good fault models for analog components such as the stuck-at-one and stuck-at-zero fault models which are widely accepted by the digital testing.

(3) *Measurement*: An analog system usually has only a few nodes accessible for measurement and testing. Without breaking connection, it is difficult to measure currents.

(4) *Nonlinear nature*: If a parameter value changes by a certain factor, the responses do not change by the same factor, i.e., the relationship between network responses and component characteristics is nonlinear, even though the circuit may be linear.

The nonlinear nature of the diagnosis problem can be observed from the following discussion. Consider a simple linear equation [26]:

$$Px = b \tag{1.1}$$

$$y = Cx \tag{1.2}$$

where b is the input-vector, y is the output-vector, x is the internal-vector, and P and C are compatible matrices. Suppose that P is changed to $P+\Delta P$, with b fixed, x will change to $x+\Delta x$ accordingly, i.e.,

$$(P+\Delta P)(x+\Delta x) = b \tag{1.3}$$

Hence,

$$\Delta x = (P+\Delta P)^{-1}b - P^{-1}b \tag{1.4}$$

and

$$\Delta y = \phi(\Delta P; P, b, C) = C(P+\Delta P)^{-1}b - CP^{-1}b \tag{1.5}$$

The problem of solving for Δy from a given ΔP is a well-known *sensitivity* problem. Conversely, solving for ΔP from a given Δy is called the *diagnosis* problem. Because the above diagnosis function ϕ may be highly nonlinear even for this simple linear equation, the inherent difficulty of the diagnosis problem is readily apparent.

Let n be the size of the system and m be the number of test points, the dimensions of ΔP and Δx are thus $n \times n$ and $m \times 1$, respectively. An important feature in the diagnosis equation (1.5) is that $m \ll n$, that is, the number of equations is much less than the number of unknowns. This leads to the difficulty of solving the diagnosis equation.

1.2 CLASSIFICATION AND COMPARISON

In order to solve fault diagnosis problems, various fault diagnosis algorithms have been proposed. These diagnosis algorithms can be roughly classified into two categories [36]:

(1) Pre-Test Simulation Technique,

(2) Post-Test Simulation Technique.

The pre-test simulation technique requires the simulation of various possible faults and the storage of the results as a dictionary [12]. In other words, for each possible ΔP, corresponding Δy is calculated and stored in a dictionary. The faulty network responses measured are compared with the dictionary entries and the closest entry to the responses by a certain measure determines the possible fault. This technique is commonly employed in digital testing and is characterized by minimal on-line computational requirements. However, the high cost of analog circuit simulation coupled with the large number of potential fault models limits the applicability of this algorithm.

The post-test simulation technique is to solve the diagnosis equation for ΔP by either increasing the number of equations or decreasing the number of unknowns. It can be further divided into simulation with sufficient measurement and simulation with failure bounds. The former is to increase the number of equations by obtaining sufficient measurements from various test points [34, 40, 42]. The latter is to decrease the number of unknowns by first assuming some components that are good and verifying them later [15, 43, 48, 49, 50].

Since sufficient measurements are needed in the former approach, multiple frequencies are applied and extremely expensive on-line computation is required to solve the high complexity of the nonlinear diagnosis equation. In contrast, the latter approach requires only a single frequency for the measurement and performs a simple on-line computation.

To evaluate a practical analog diagnosis algorithm, the following criteria are concerned [36]: computational requirements, number of test points and test vector employed, robustness to tolerance effects, availability of models, and the degree to which the algorithm is amenable to parallel processing. Based on the above criteria, the Post-Test Simulation with Failure Bounds is better than others in test application.

1.3 MOTIVATION

The research of analog fault diagnosis in the past years has posed several important and challenging problems [26]. First, a component tolerance problem is concerned. Some proposed algorithms work well for the case that the nonfault components assume their nominal value exactly. However, the values of nonfault components in a practical circuit may deviate from their nominal value within a predefined tolerance. As a result, the fault components may not be properly located or even be detected. This motivates the study of an algorithm that can alleviate the error effect due to component tolerance.

With the rapidly increasing complexity and size of modern electronic systems, the ability to adequately design a diagnosable system is a prime requisite for rapid fault location. Some fault location and prediction algorithms can precisely locate and predict faulty components for small size networks. Due to the substantial increase of on-line computation for a large network, however, direct implementation of such algorithms for a practically large circuit network would be impractical. In practice, a faulty network generally has failures in a small portion of the network and the remaining parts are fault-free. Decomposing a large circuit network into several small subnetworks seems a very attractive idea, and, thus, serves as motivation to study a decomposition approach fault diagnosis and fault prediction algorithm for large analog circuit networks.

Usually, more test points can faster and more precisely locate faults, but, unfortunately, modern electronic systems are often multi-layered and/or coated, thereby limiting the accessibility of test points which are available at the externally accessible terminals of a printed circuit board. As the number of components in a unit increases, it is difficult to provide proportionately more I/O terminals. It limits the implementation of many fault diagnosis algorithms. In order to provide sufficiently more test points while still keeping low pin overhead for analog circuit testing, an analog Built-in Self-test structure (ABIST) is motivated.

Although an ABIST structure may provide as many test points as required for a circuit under test, the number of test points should not be indefinitely increased because of the placement and routing problems. Therefore, it is necessary to select an appropriate minimum set of test points that is sufficient to diagnose a circuit, as well as to develop an efficient test point selection algorithm.

1.4 THESIS ORGANIZATION

This dissertation is organized as follows. In the next chapter, a fault location algorithm is presented. Basically, faults are located by checking the consistency or inconsistency of a set of linear equations. An Analog Automatic Test Program Generator (AATPG) software package is developed for fault diagnosis use. Some examples are presented which demonstrate that the proposed algorithms can precisely locate faults in analog circuits.

In Chapter 3, a fault prediction approach is proposed. For the case that the parameters of a potential faulty component are assumed to change gradually during each maintenance period, by continuously monitoring the responses of the network, the proposed fault prediction algorithm can predict whether or not any of the network components is about to fail. The predicted fault components are then replaced before the system actually fails.

In Chapter 4, a decomposition approach fault diagnosis and fault prediction algorithm for large circuit network is addressed. Basically, a large circuit network is decomposed into several small subnetworks. Also the error effect due to component tolerance in practical circuits can be alleviated and faults in large circuit networks can be precisely predicted. Also this approach can be extend for nonlinear circuits.

In Chapter 5, an analog Built-in Self-test (ABIST) structure is proposed. This structure not only provides more test points for analog circuit testing, but also acquires test data at various test points simultaneously. The detailed structure and its operations

are described, the simulation result and hardware implementation of the design are presented, and the VLSI implementation of the ABIST structure is discussed.

In Chapter 6, testability design principle is first outlined. Based on the design principle, the issues of diagnosability and test set generation are addressed.

Finally, the last chapter summarizes the work of this dissertation research and presents suggestions for related future research.

CHAPTER 2

FAULT DIAGNOSIS

Fault diagnosis includes both fault detection and fault location. Fault detection is obviously a minimum requirement for fault location. When a circuit fails to a functional test, a diagnostic test is carried out to identify faulty components and to evaluate the faulty component values for studying the cause of failure. The information is fed back to the designer for further improvement. In this chapter, the emphasis is placed on the fault location of analog circuit networks. A diagnosis equation is derived first, followed by fault location and fault evaluation approach. An efficient fault location algorithm is presented and an AATPG that generates test programs for fault diagnosis use is discussed. Some examples are presented which demonstrate that the proposed algorithms can precisely locate faults in analog circuits.

2.1 DIAGNOSIS EQUATION

Based on Tellegen theorem [13] and the concept of the adjoint network [7], a diagnosis equation describing the relationship between the variation of component parameter and the change of voltage at accessible nodes is derived for a linear network. Consider a linear network, N, that has b internal branches and m accessible terminals. Let $\bar{N}$ and $\hat{N}$ be a faulty network and the adjoint network of N, respectively. The adjoint network $\hat{N}$ is topologically identical to the original network N [7].

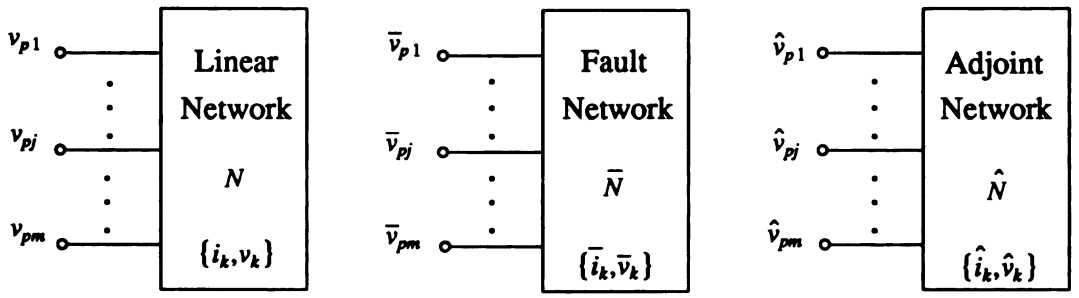


Figure 2.1 Circuit Networks.

Based on Tellegen theorem, we have

$$\text{For } N: \quad \sum_{k=1}^{b} (\hat{v}_k i_k - v_k \hat{i}_k) = -\sum_{j=1}^{m} (\hat{v}_{pj} i_{pj} - v_{pj} \hat{i}_{pj}) \tag{2.1}$$

$$\text{For } \bar{N}: \quad \sum_{k=1}^{b} [\hat{v}_k (i_k + \Delta i_k) - (v_k + \Delta v_k) \hat{i}_k] = -\sum_{j=1}^{m} [\hat{v}_{pj} (i_{pj} + \Delta i_{pj}) - (v_{pj} + \Delta v_{pj}) \hat{i}_{pj}] \tag{2.2}$$

Subtracting Equation (2.2) from Equation (2.1), we get

$$\sum_{k=1}^{b} (\hat{v}_k \Delta i_k - \Delta v_k \hat{i}_k) = -\sum_{j=1}^{m} (\hat{v}_{pj} \Delta i_{pj} - \Delta v_{pj} \hat{i}_{pj}) \tag{2.3}$$

With branch constraints

$$i_k = y_k v_k, \quad i_k + \Delta i_k = (y_k + \Delta y_k)(v_k + \Delta v_k), \quad \text{and} \quad \hat{i}_k = y_k \hat{v}_k,$$

Equation (2.3) yields

$$\sum_{k=1}^{b} \hat{v}_k \Delta x_k = \Delta p \tag{2.4}$$

where

$$\Delta x_k = \Delta y_k (v_k + \Delta v_k) = \Delta y_k \bar{v}_k, \tag{2.5}$$

$$\Delta p = \sum_{j=1}^{m} (\Delta v_{pj} \hat{i}_{pj} - \hat{v}_{pj} \Delta i_{pj}) \tag{2.6}$$

A generalized form of Equation (2.4) can be expressed as

$$\sum_{k=1}^{b} w_k \Delta x_k = \Delta p \tag{2.7}$$

where w_k and Δx_k are listed in Table 2.1.

Table 2.1 w_k and Δx_k of Different Types of Components

Component type		w_k	Δx_k
admittance	y	$\hat{v}_k$	$\Delta y_k(v_k+\Delta v_k)$
controlled sources	cccs	$\hat{v}_k^h$	$\Delta\beta_k(i_k^j+\Delta i_k^j)$
	vccs	$\hat{v}_k^h$	$\Delta g_k(v_k^j+\Delta v_k^j)$
	ccvs	$\hat{i}_k^h$	$\Delta\gamma_k(i_k^j+\Delta i_k^j)$
	vcvs	$\hat{i}_k^h$	$\Delta\mu_k(v_k^j+\Delta v_k^j)$

(Note: superscripts h and j represent controlled and controlling branches of a controlled source, respectively.)

Consider a matrix form for Equation (2.7)

$$W\Delta X = \Delta P \tag{2.8}$$

where

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1b} \\ w_{21} & w_{22} & \cdots & w_{2b} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ w_{m1} & w_{m2} & \cdots & w_{mb} \end{bmatrix}$$

$$\Delta X = [\Delta x_1 \ \Delta x_2 \cdots \Delta x_b]^T$$

$$\Delta P = [\Delta p_1 \Delta p_2 \cdots \Delta p_m]^T$$

Note that $w_{ij} = \hat{v}_j^{(i)}$ represents the calculated voltage of the jth branch of the adjoint network when a unity current excitation is applied at the ith accessible terminal, and $\Delta p_j = \Delta v_{pj} = \overline{v}_{pj} - v_{pj}$ is the voltage difference at the j-th terminals of both faulty and nonfaulty networks.

2.2 FAULT LOCATION

Generally, the number of components in a circuit is always much more than the number of test points, i.e., $b \gg m$. As a result, there is no unique nontrivial solution for (2.8). In practice, however, it is reasonable to assume that there are only a few faulty components in a reliably designed network. Of course, we do not know how many faulty components exist, or where they are located in advance.

Let f be the number of faulty components. Without loss of generality, we assume that the first f components are faulty and the others are nonfaulty, i.e., $\Delta y_k = 0$, or $\Delta x_k = 0$, for $k = f+1, \cdots, b$. Therefore, Equation (2.8) can be reduced as

$$W_f \Delta X_f = \Delta P \tag{2.9}$$

where

$$W_f = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1f} \\ w_{21} & w_{22} & \cdots & w_{2f} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ w_{m1} & w_{m2} & \cdots & w_{mf} \end{bmatrix}$$

$$\Delta X_f = [\Delta x_1 \ \Delta x_2 \cdots \Delta x_f]^T$$

Equation (2.9) has a unique nontrivial solution only if

$$rank[W_f \mid \Delta P] = rank[W_f] = f \tag{2.10}$$

The condition (2.10) determines the number of faulty components and also identifies the faulty components. The detailed fault location process is described in Algorithm I.

2.3 FAULT EVALUATION

Once the faulty components are identified, we may evaluate the faulty component values to study the cause of the failures. Suppose that a matrix W_f of the corresponding faulty components satisfies condition (2.10), with the existence of the generalized inverse of W_f, Equation (2.9) yields

$$\Delta X_f = (W_f^T W_f)^{-1} W_f^T \Delta P \tag{2.11}$$

which gives the values of Δx_k, for k=1,2, $\cdots$,f. The next step is to find all node voltages.

Suppose that the admittance of a faulty component is changed from y to $y+\Delta y$. This is equivalent to y connecting in parallel with Δy. According to the substitution theorem [13], Δy can be substituted by a current source Δx, which depends on the deviation Δy and its branch voltage, i.e., $\Delta x_k=\Delta y_k(v_k+\Delta v_k)$. Similarly, the deviation of the controlled sources can be substituted by a source, referred to as *substituted source*, as illustrated in Table 2.2. Thus, a faulty network is equivalent to the corresponding nonfaulty network N connecting with some substituted sources Δx_k, as shown in Figure 2.2.

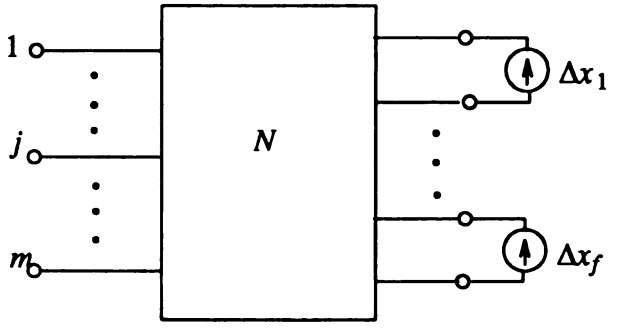
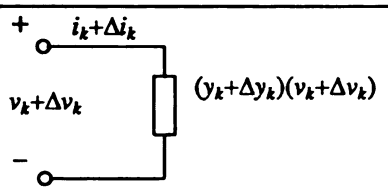
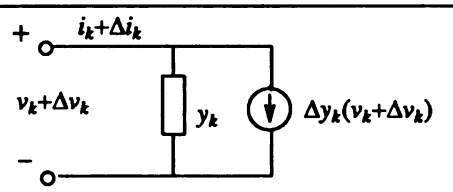
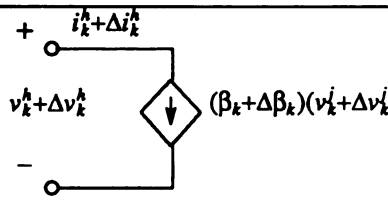
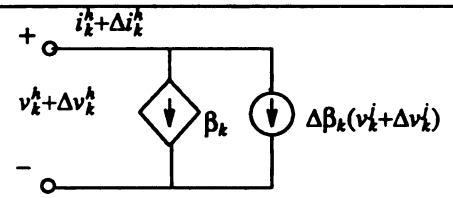
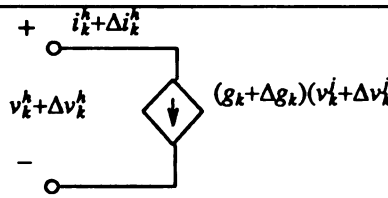
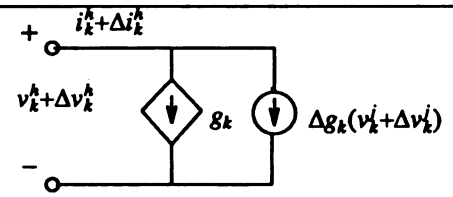
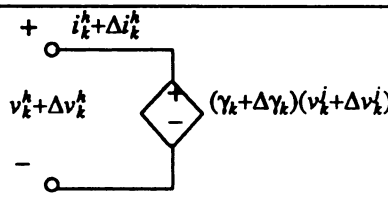
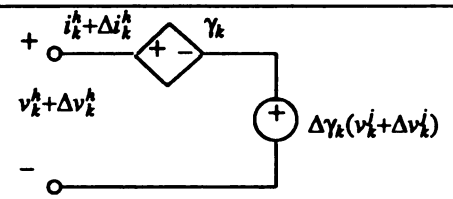
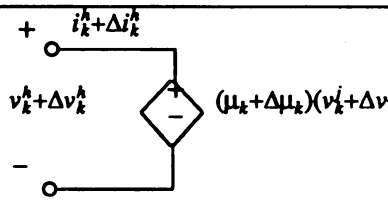
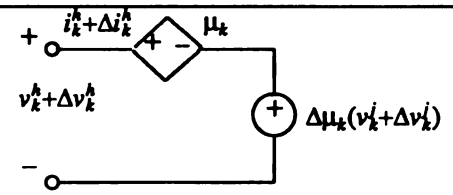


Figure 2.2 Equivalent Faulty Network.

Table 2.2 Substituted Sources of Different Types of Components

type	faulty components	substituted sources
y		
CCCS		
VCCS		
CCVS		
VCVS		

For a nonfaulty network N the nodal equation is expressed as follows

$$[H]\begin{bmatrix} V_N \\ I_N \end{bmatrix} = [I_{sn}] \tag{2.12}$$

where H is the nodal admittance matrix, V_N is the node voltage vector, I_N is the current vector of the voltage controlled sources, and I_{sn} is the node current source vector. Similarly, the nodal equation of the faulty network of Figure 2.2 is expressed as

$$[H]\begin{bmatrix} \bar{V}_N \\ \bar{I}_N \end{bmatrix} = [I_{sn}] - [\Delta X_f] \tag{2.13}$$

where $\bar{V}_N$ and $\bar{I}_N$ are the node voltage and current vectors of the faulty network, where ΔX_f is the substituted source vector. Since the matrix H, vectors I_{sn}, and ΔX_f are all known, Equation (2.13) can be solved for the vector $\bar{V}_N$ and $\bar{I}_N$. Once both $\bar{V}_N$ and $\bar{I}_N$ are computed, the faulty component deviation Δe_f can be evaluated by ΔX_f.

2.4 EQUIVALENT FAULT SETS

The number of faulty components and the faulty components can be determined by checking Equation (2.10). If a set of components satisfies the condition in Equation (2.10), the sets of components are referred to as *fault set candidates*. In practice, however, more than one candidate may be concluded. Moreover, only one candidate that contains all faulty components is the *actual fault set*, and the remaining candidates, referred to as *equivalent fault sets*, that contain some nonfaulty components, but they produce the same responses as the actual fault set at the accessible terminals. In other words,

$$rank[W_f \mid \Delta P] = rank[W_{f(act)} \mid \Delta P] = f$$

The existence of the equivalent fault sets is attributed to either the topological structure, or test points. The details are shown in the following properties.

Property 2.1

If the actual fault set contains a subset that consists of either

a) l components in loop L which consists of $l+1$ components, or

b) l components in cutset C which consists of $l+1$ components incident to an inaccessible node,

then an f-component-set formed by a combination of any l components of loop L or cutset C together with other $f-l$ components of the actual fault set, is an equivalent fault set.

Property 2.2

If $m-f+1$ row dependence of W_f is consistent with row dependence of $W_{f(act)}$, and $rank[W_f] = f$, the f-component-set corresponding to W_f is an equivalent fault set.

The proofs of Properties 2.1 and 2.2 are shown in Appendix 1.

Recall that, in the fault location process, Equation (2.9) is used to estimate the number of faulty components and to evaluate the faulty component values. The underlying assumption for that process is $\Delta x_k=0$, for $k=k+1, \cdots, b$, i.e., the components that are not in the fault set under processing, are faulty-free. The assumption is true only if the actual fault set is processed. Therefore, the evaluated values of the components in the actual fault set are virtually the same irrespective of the locations of the external excitations and the applied frequencies. However, the assumption is not true when the equivalent fault set is processed, because some Δx_k's are not zeros. As a result, with the application of different frequencies or different locations of the excitations, the estimated values of the components in equivalent fault sets will be different. Based on this concept, we may identify the actual fault set either by applying a different frequency, or by relocating the external excitation for fault diagnosis use. In other words, if more than one fault sets are obtained in the fault location process, properly relocating the excitation is needed. From the component values estimated under two different excitations, we should be able to identify the actual fault set.

2.5 SOFTWARE IMPLEMENTATION

The fault location algorithm discussed above is summarized in Algorithm I.

Algorithm I.

(off-line phase)

Step 1. Input the network topology and its component nominal values to construct the nodal admittance matrix H.

Step 2. Apply a unit current excitation to the ith accessible terminal of an adjoint network. Construct the matrix W by the computer branch voltages $\hat{v}_j^{(i)}$ (j=1,2, $\cdots$,b, i=1,2, $\cdots$,m) of adjoint networks

Step 3. Calculate the voltage V at the accessible terminals of the fault-free network N.

(on-line phase)

Step 1. Obtain the voltages $\bar{V}$ measured at the accessible terminals of a faulty network, (either from the ATE or by simulation.)

Step 2. Load the data generated in the off-line phase and initialize r=1.

Step 3. *Repeat*

3.1. pick up a combination of r columns of matrix W to form W_r;

3.2. Check if $rank\,[W_r \mid \Delta P] = rank[W_r] = r$;

3.3.1. Calculate the following equations for the actual component value e_k;

(a). $\Delta X_f = (W_f^T W_f)^{-1} W_f^T \Delta P$

(b). $[H][\bar{V}_n]=[I_{sn}]-[\Delta X_f]$

(c). $\Delta y_k(\bar{v}_k) = \Delta x_k$, where k=1,2, $\cdots$,f

(d). $e_k = y_k+\Delta y_k$, where k=1,2, $\cdots$,f

3.3.2. If $e_k>0$, record this fault set; *Until* all combinations are applied.

Step 4. If no fault set is identified, then $r=r+1$, and Go To Step 3.

Step 5. If only one possible fault set is concluded, then faulty elements are located, and the process is terminated.

Step 6. (more than one possible fault sets)

Apply the second excited source, and input the voltages measured at the accessible terminals of a faulty network.

Repeat

6.1. Take a fault set at a time;

6.2. Calculate the equations (a) to (d) of step 3.3.1 for e_k;

6.3. If $e_k<0$, eliminate this set;

Until all fault sets are applied.

Step 7. Compare two e_k's obtained from Steps 3.3 and 6.2 for each fault set. If they are equal, then this set is the actual fault set.

An Analog Automatic Test Program Generator (AATPG) that generates test programs for fault diagnosis use has been developed based on Algorithm I. The process is divided into two phases: off-line and on-line [23].

The off-line phase, corresponding to the test system design stage, is used by the test system designer to input nominal network specifications and generate a data base which is used by the on-line phase. The input requirements in the test program generation are circuit description, input frequency, and accessible test terminals. During the on-line phase, the data generated by off-line phase (the admittance matrix H, the matrix W, and the accessible terminal voltage V) are loaded. The test data are acquired either by fault simulation or from ATE (Automatic Test Equipment). With the measured voltages, Algorithm I is carried out to identify the faulty components and evaluate the component values.

In both off-line and on-line phases, AATPG provides user-oriented interfaces to simplify the process of generating a new test program. The AATPG has been implemented on VAX 8600 (ULTRIX) in Fortran and C [23].

2.6 EXAMPLES

In order to demonstrate the effectiveness of the proposed fault location algorithm, two examples are given.

Example 2.1

Consider a linear resistive network, as shown in Figure 2.3, consisting of all unity resistances. Suppose that nodes 1, 6 and 7 are taken as the test points.

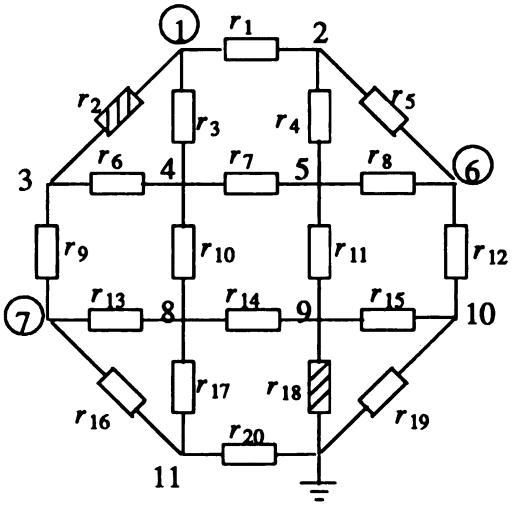


Figure 2.3 The Resistor Network in Example 2.1.

In the off-line phase, the *W*-matrix and the voltages that are measured at the test points when the network is fault-free, must be established before the test is conducted. According to the schematic circuit diagram of Figure 2.3, the *H*-matrix is generated as follows

$$H = \begin{bmatrix} 3 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 3 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 3 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 3 \end{bmatrix}$$

Taking the H-matrix, the W-matrix is constructed as follows. We first apply a unity excitation current to the node 1 of the adjoint network $\hat{N}$, and then calculate the branch voltages of the network $\hat{N}$, which is listed in the first column of W-matrix in Table 2.3. The remaining columns of the W-matrix are formed in a similar way. The computed voltage vector V for the nominal network is shown in the second column of Table 2.4.

When the test is being conducted, the measured test data may be obtained from an automatic test equipment (ATE). In this moment, however, the ATE is not available and thus a set of simulated test data is employed. In this example, we assume that the components 2 and 18 are faulty with r_2=0.4Ω and r_{18}=1.5Ω. Column 3 and 4 of Table 2.4 list the measured test data $\overline{V}$ and the voltage difference, $\Delta P = V - \overline{V}$, respectively.

Table 2.3 *W* Matrix of Example 2.1

element	(1)	(6)	(7)
1	3.5000*e*–01	–1.3750*e*–01	1.1250*e*–01
2	3.0000*e*–01	8.3333*e*–02	–1.1667*e*–01
3	3.5000*e*–01	5.4167*e*–02	4.1667*e*–03
4	1.5000*e*–01	7.9167*e*–02	2.9167*e*–02
5	2.0000*e*–01	–2.1667*e*–01	8.3333*e*–02
6	5.0000*e*–02	–2.9167*e*–02	1.2083*e*–01
7	1.5000*e*–01	–1.1250*e*–01	1.3750*e*–01
8	5.0000*e*–02	–2.9583*e*–01	5.4167*e*–02
9	2.5000*e*–01	1.1250*e*–01	–2.3750*e*–01
10	2.5000*e*–01	1.3750*e*–01	–1.2500*e*–02
11	2.5000*e*–01	2.6250*e*–01	1.1250*e*–01
12	2.5000*e*–01	4.8750*e*–01	1.3750*e*–01
13	5.0000*e*–02	–4.1666*e*–03	3.4583*e*–01
14	1.5000*e*–01	1.2500*e*–02	2.6250*e*–01
15	5.0000*e*–02	–7.0833*e*–02	7.9167*e*–02
16	2.0000*e*–01	1.1667*e*–01	4.1667*e*–01
17	1.5000*e*–01	1.2083*e*–01	7.0833*e*–02
18	3.5000*e*–01	3.4583*e*–01	2.9583*e*–01
19	3.0000*e*–01	4.1667*e*–01	2.1667*e*–01
20	3.5000*e*–01	2.3750*e*–01	4.8750*e*–01

Table 2.4 Simulation Data of Example 2.1

Node	V	$\bar{V}$	ΔP
1	1.1000e+00	1.0750e+00	−2.5026e−02
6	5.5000e−01	5.7751e−01	2.7505e−02
7	5.5000e−01	6.1960e−01	6.9599e−02

Since we do not know how many faulty components exist, we will first check the number of faulty components. Specifically, we start with f=1 for Equation (2.10) and find that the following matrix satisfies Equation (2.10) when f=2,

$$[W_f, \Delta P] = \begin{array}{c} \\ 1 \\ 6 \\ 7 \end{array} \begin{bmatrix} 2 & 18 & \Delta P \\ 3.0000e{-}01 & 3.5000e{-}01 & -2.5026e{-}02 \\ 8.3333e{-}02 & 3.4583e{-}01 & 2.7505e{-}02 \\ -1.1667e{-}01 & 2.9583e{-}01 & 6.9599e{-}02 \end{bmatrix}$$

Since the fault set {2,18} is the only candidate, the components 2 and 18 are identified as faulty.

The values of the faulty component 2 and 18 can be computed as follows. First, Equation (2.11) is used to calculate ΔX_f, which is listed in the second column of Table 2.5. Then, two substituted current sources ΔX_f, i.e., Δx_2 and Δx_{18}, are connected in parallel with corresponding components 2 and 18, as shown as Figure 2.4. The Vectors $\bar{V}_N$ and $\bar{I}_N$ of the substituted network are computed from Equation (2.13). In this example, the branch voltage vector $\bar{V}_N$ for components 2 and 18 are listed in the third column of Table 2.5. The computed branch voltages are used to evaluate the component deviation, i.e., $\Delta y = \Delta x_k / \bar{v}_k$, k=2,18. Finally, the faulty component values are determined by $e_k = y_k + \Delta y_k$, or $r_2 = 1/e_k = 0.4$ and $r_{18} = 1.5$.

Table 2.5 Component Evaluation

Component	ΔX_f	branch voltage	Δy_k	$r_k=\frac{1}{e_k}$
2	2.4512e-01	1.6341e-01	1.5000e+00	4.0000e-01
18	-1.3860e-01	4.1579e-01	-3.3334e-01	1.5000e+00

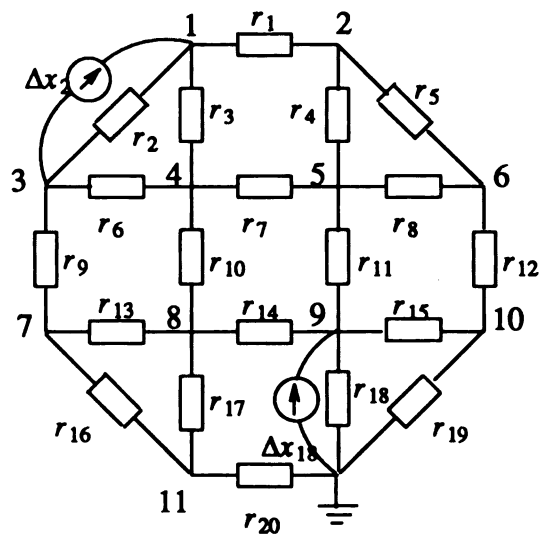


Figure 2.4 Equivalent Faulty Network in Figure 2.3.

Example 2.2

Consider a linear active circuit with a controlled source, as shown in Figure 2.5. The component values are listed in Table 2.6, and the nodes 1, 2, 4 and 5 are taken as the test points.

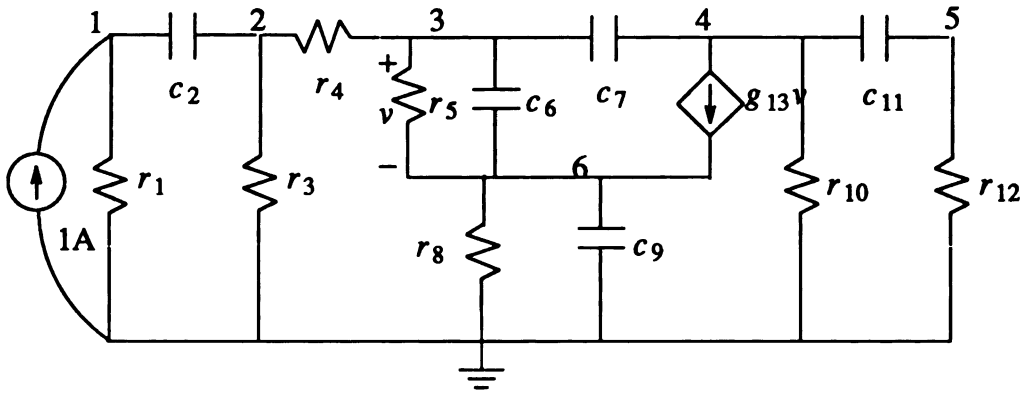


Figure 2.5 An Active Circuit.

Table 2.6 Component Values of Example 2.2

Component	Value	Component	Value
r_1	20Ω	r_8	30Ω
c_2	20μF	c_9	10μF
r_3	75Ω	r_{10}	10Ω
r_4	10Ω	c_{11}	20μF
r_5	40Ω	r_{12}	10Ω
c_6	15μF	g_{13}	10
c_7	25μF		

Table 2.7 *W* Matrix of Example 2.2

element	(1)	(2)	(4)	(5)
1	(6.1229e+00 -1.3835e-02)	(6.1229e+00 -8.3133e-03)	(1.2229e-03 -1.6108e-02)	(1.2357e-03 -1.6107e-02)
2	(5.7220e-06 -5.5215e-03)	(3.3379e-06 2.4362e-03)	(6.4091e-06 4.8615e-07)	(6.4089e-06 4.9174e-07)
3	(6.1229e+00 -8.3133e-03)	(6.1229e+00 -1.0750e-02)	(1.2164e-03 -1.6109e-02)	(1.2293e-03 -1.6108e-02)
4	(6.1221e+00 7.8425e-03)	(6.1221e+00 5.4066e-03)	(-7.7410e-04 1.0202e-02)	(-7.8222e-04 1.0201e-02)
5	(7.2482e-04 -6.4148e-03)	(7.2227e-04 -6.4151e-03)	(1.8619e-03 -1.0403e-02)	(1.8702e-03 -1.0401e-02)
6	(7.2482e-04 -6.4148e-03)	(7.2227e-04 -6.4151e-03)	(1.8619e-03 -1.0403e-02)	(1.8702e-03 -1.0401e-02)
7	(-2.0568e-05 -1.0487e-06)	(-2.0569e-05 -1.0394e-06)	(-4.1600e-05 6.3636e-03)	(-4.6664e-05 6.3636e-03)
8	(6.9070e-05 -9.7410e-03)	(6.5195e-05 -9.7411e-03)	(1.2868e-04 -1.5908e-02)	(1.4134e-04 -1.5908e-02)
9	(6.9070e-05 -9.7410e-03)	(6.5195e-05 -9.7411e-03)	(1.2868e-04 -1.5908e-02)	(1.4134e-04 -1.5908e-02)
10	(8.1446e-04 -1.6155e-02)	(8.0803e-04 -1.6155e-02)	(2.0322e-03 -3.2674e-02)	(2.0582e-03 -3.2672e-02)
11	(-1.2855e-05 -6.5751e-07)	(-1.2855e-05 -6.5379e-07)	(-2.6000e-05 -1.6354e-06)	(-3.2331e-05 7.9561e-03)
12	(8.2731e-04 -1.6154e-02)	(8.2089e-04 -1.6154e-02)	(2.0582e-03 -3.2672e-02)	(2.0905e-03 -4.0628e-02)
13	(7.4539e-04 -6.4137e-03)	(7.4284e-04 -6.4140e-03)	(1.9035e-03 -1.6766e-02)	(1.9168e-03 -1.6765e-02)

Similar to Example 2.1, the W-matrix is computed as shown in Table 2.7, and the voltage vector V is listed in the second column of Table 2.8. If we assume that the components r_4 and g_{13} are faulty with r_4=20Ω and g_{13}=50, the faulty components are located in accordance with the measured voltage vector $\bar{V}$ and ΔP shown in Table 2.8.

Table 2.8 Simulation Data of Example 2.2

Node	V	$\bar{V}$	ΔP
1	(6.1229e+00,−1.3835e−02)	(8.8244e+00,−7.4149e−03)	(2.7015e+00,6.4199e−03)
2	(6.1229e+00,−8.3133e−03)	(8.8244e+00,−2.9683e−03)	(2.7015e+00,5.3449e−03)
4	(1.2229e−03,−1.6108e−02)	(3.1670e−03,−1.0020e−02)	(1.9442e−03,6.0881e−03)
5	(1.2357e−03,−1.6107e−02)	(3.1750e−03,−1.0018e−02)	(1.9393e−03,6.0896e−03)

Table 2.9 illustrates all fault set candidates that satisfy Equation (2.10). In order to precisely identify the actual fault set, the unity excitation current is relocated at node 5. Table 2.9 lists the evaluated component values for these two excitations. From the differences listed in Table 2.9, the actual fault set {4,13} is identified, i.e., the components 4 and 13 are known as faulty with r_4=1/0.05=20, and g_{13}=50, which are consistent with the given simulated fault values.

Table 2.9 Simulation Results of Example 2.2

Fault set	Fault values evaluated		
	Excitation at node 1	Excitation at node 5	Difference
3 4	(1.9341e−02,−1.0381e−02) (4.4008e−02, 1.0394e−02)	(2.0956e+01, 7.5264e+00) (1.8184e+01, 3.0937e+00)	9.9934e−01
4 5	(5.0006e−02, 2.47396e−06) (5.8136e+01,−1.8496e+01)	(5.0627e−02,−8.8866e−03) (5.8135e+01,−1.8504e+01)	1.7336e−01
4 6	(5.0006e−02, 2.4740e−06) (9.2486e+00, 1.2056e+01)	(5.0627e−02,−8.8866e−03) (9.2485e+00, 1.2055e+01)	1.7336e−01
4 7	(4.9992e−02,−5.5611e−06) (1.0575e+00, 2.2879e+00)	(4.2446e−02, 1.0026e−02) (2.0806e+01, 1.2509e+01)	9.1597e−01
4 8	(5.0005e−02, 3.3373e−06) (2.9067e+01, 5.6407e+00)	(5.2221e−02,−8.8738e−03) (2.9068e+01, 5.6390e+00)	1.7273e−01
4 9	(5.0005e−02, 3.3373e−06) (4.6209e+00, 1.0898e+01)	(5.2221e−02,−8.8738e−03) (4.6211e+00, 1.0897e+01)	1.7273e−01
4 10	(5.0003e−02, 1.4904e−06) (7.9698e+00, 1.9130e+00)	(5.1185e−02,−4.1434e−03) (6.2489e+00, 1.9185e+00)	2.0996e−01
4 13	(5.0000e−02, 1.8072e−10) (5.000e+01, 3.48859e−03)	(5.0000e−02, 3.7433e−08) (5.000e+01,−1.5038e−04)	8.2385e−05

CHAPTER 3

FAULT PREDICTION

For the case that the nonfaulty components assume their nominal value exactly, referred to as the *ideal case*, the proposed fault location algorithm can precisely locate faults. On the other hand, for the *practical case* that the nonfaulty component values may deviate from their nominal values with a predefined component tolerance, the algorithm still can locate the most of the faults. In some situations, however, the deviation may affect the determination of the system status even though all components are fault-free. This accumulation is referred to as the *error effect*. In the ideal case, the error effect is merely the truncation error of the computer. In the practical case, however, the error effect highly depends on the predefined deviation percentage of the nonfaulty components.

Our study has found that if the predefined deviation percentage is within 5%, the proposal location algorithm can confidently locate those (soft) faulty components whose values are deviated, at least, 50% of the nominals. When the fault deviation is less than 50%, however, it is really difficult to determine whether the system failed. Moreover, if we assume that the parameters of a potential faulty component are changed gradually, the faulty components can then be detected after system failed. For a reliable design, the potential faulty components should be identified as soon as possible and replaced before the system fails. Therefore, a fault prediction algorithm is motivated.

Of course, not all failures can be predicted. For example, there is little hope of predicting component failure due to random external effects (improper operation, lightning, etc.). However, there is a significant experimental evidence to suggest that when electronic components fail due to permanent overstress (high temperature, continued overload operation, material fatigue, etc.), their parameters change sufficiently slowly thus enabling a prediction of the time at which they will go out of tolerance by statistical trending techniques [2,37].

3.1 ERROR EFFECT ANALYSIS FOR COMPONENT TOLERANCE

Consider a faulty network $\bar{N}'$, whose nonfaulty component values deviate within a prescribed tolerance. It is shown in Figure 3.1 along with corresponding nonfaulty network N and adjoint network $\hat{N}$.

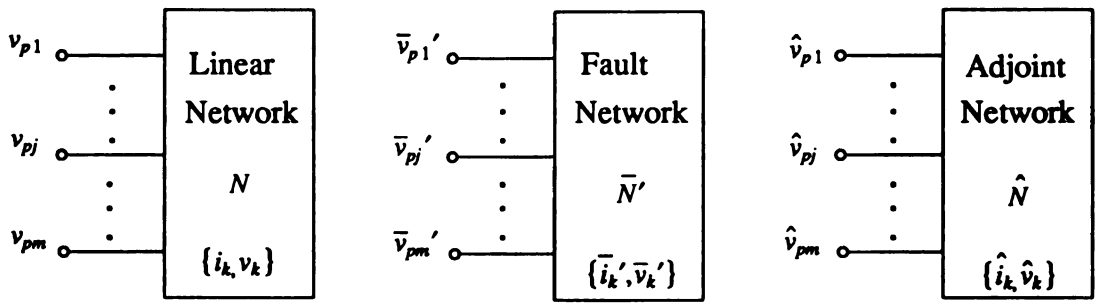


Figure 3.1 Circuit Networks.

Based on Tellegen theorem for $\bar{N}'$, we have

$$\sum_{k=1}^{b} [\hat{v}_k(i_k+\Delta\bar{i}_k')-(v_k+\Delta\bar{v}_k')\hat{i}_k] = -\sum_{j=1}^{m} [\hat{v}_{pj}(i_{pj}+\Delta\bar{i}_{pj}')-(v_{pj}+\Delta\bar{v}_{pj}')\hat{i}_{pj}] \tag{3.1}$$

Subtracting Equation (3.1) from Equation (2.1), we get

$$\sum_{k=1}^{b} (\hat{v}_k\Delta\bar{i}_k'-\Delta\bar{v}_k'\hat{i}_k) = -\sum_{j=1}^{m} (\hat{v}_{pj}\Delta\bar{i}_{pj}'-\Delta\bar{v}_{pj}'\hat{i}_{pj}) \tag{3.2}$$

If Δy_k and $\Delta y_k'$ are the deviations due to fault and component tolerance, respectively, then, with the branch constraints for $\overline{N}'$

$$i_k+\Delta\bar{i}_k' = (y_k+\Delta\bar{y}_k')(v_k+\Delta\bar{v}_k')$$

where $\Delta\bar{y}_k' = \Delta y_k+\Delta y_k'$, the left hand of Equation (3.2) is expressed as

$$\sum_{k=1}^{b}(\hat{v}_k\Delta\bar{i}_k'-\Delta\bar{v}_k'\hat{i}_k) = \sum_{k=1}^{b}(\hat{v}_k\Delta\bar{y}_k'\,\bar{v}_k') = \sum_{k=1}^{b}\hat{v}_k(\Delta y_k+\Delta y_k')\bar{v}_k'$$

$$= \sum_{k=1}^{b}(\hat{v}_k\Delta y_k\bar{v}_k)+\sum_{k=1}^{b}[\hat{v}_k\Delta y_k'\,\bar{v}_k'+\hat{v}_k\Delta y_k(\bar{v}_k'-\bar{v}_k)]$$

Equation (3.2) yields

$$\sum_{k=1}^{b}\hat{v}_k\Delta x_k = \Delta p'-\Delta e' \tag{3.3}$$

where

$$\Delta x_k = \Delta y_k\bar{v}_k$$

$$\Delta p' = \sum_{j=1}^{m}(\Delta\bar{v}_{pj}'\hat{i}_{pj}-\hat{v}_{pj}\Delta\bar{i}_{pj}') \tag{3.4}$$

$$\Delta e' = \sum_{k=1}^{b}(\hat{v}_k\Delta y_k'\,\bar{v}_k'+\hat{v}_k\Delta y_k(\bar{v}_k'-\bar{v}_k)), \tag{3.5}$$

note that $\bar{v}_k$ is the kth branch voltage of a faulty network without tolerance, and $\bar{v}_k'$ is the kth branch voltage of a faulty network with tolerance.

Comparing Equation (3.3) with (2.4) in these two different cases, the only difference is that there exists an additional term $-\Delta e'$ in the right-hand side of Equation (3.3). This term, referred to as the *error term*, is caused by the component tolerance. In the ideal case, both $\Delta y_k'$ and $(\bar{v}_k'-\bar{v}_k)$ are zero, hence $\Delta e'=0$. In practice, we must face the problem of $\Delta e'\neq 0$. If we directly implement the algorithm developed for the ideal case to practical case, the error term $\Delta e'$ will be ignored. When fault deviation is larger than 50%, that is, Δy is more than 10 times larger than $\Delta y_k'$, the ignoring of $\Delta y_k'$ may not

cause any problem. But when fault deviation is less than 50%, a large error effect may be caused so that the faulty components sometimes cannot be precisely located.

Now, let us closely examine the error term $\Delta e'$. Consider the second term of Equation (3.5), the deviation $(\overline{v}_k'-\overline{v}_k)$ due to the component tolerance is generally small for a reliable design even if the circuit fails. Thus, the second term can be neglected if a parameter of a potential faulty component is changed gradually.

The first term of Equation (3.5) can be neglected only if $\overline{v}_k'$ is very small for all branches in the given circuit. This is virtually impossible in many practical circuits. Therefore, the error effect due to the error term in Equation (3.3) cannot be eliminated and will limit the implementation of the fault location process in the practical case.

3.2 BASIC CONCEPT OF FAULT PREDICTION

Our objective is to keep the error term within a reasonably small range so that one can locate the faulty components precisely and evaluate the component values correctly.

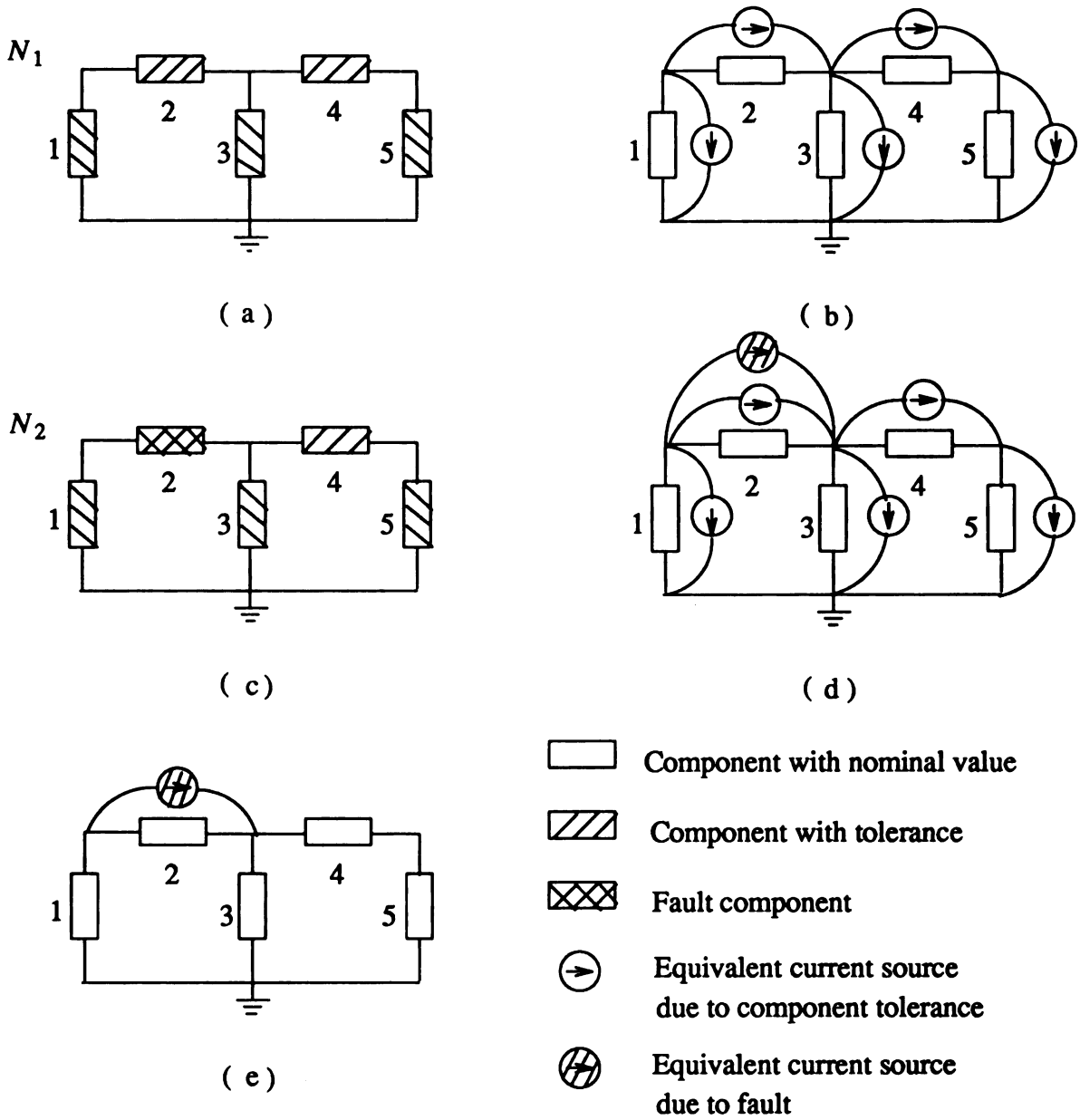


Figure 3.2 Basic Concept of Fault Prediction.

Consider a network N_1, as shown in Figure 3.2 (a). Each component in N_1 is assumed to have a predescribed component tolerance. According to the substitution theorem, the component with a tolerance can be modeled by a current source (for component tolerance) connected in parallel with the component having its nominal value. Therefore, network N_1 is modeled as shown in Figure 3.2(b). On the other hand, consider a network N_2, as shown in Figure 3.2 (c), having the same topological structure and component values as N_1, except that component #2 is assumed to be faulty. Faulty component #2 can be modeled by connecting an additional current source (for fault) in parallel with the component, as shown in Figure 3.2 (d). Suppose that the corresponding components in both networks N_1 and N_2 have the same component tolerance, which implies that the corresponding current sources modeled for component tolerance are the same. Therefore, by the superposition theorem [13], the corresponding current sources due to component tolerance can be canceled by each other. The resultant circuit (Figure 3.2 (e)) shows each component with its nominal.

In practice, it is very difficult to predict whether the corresponding components in both N_1 and N_2 have the same component tolerance. However, for a reliable circuit design, it is reasonable to assume that the nonfaulty component parameters have virtually no change during a reasonably short period of time. In other words, if a network is continuously monitored, then the component tolerance of the nonfaulty components are virtually the same. In the fault prediction approach, these two consecutive measurements obtained from the same network can be treated in the same way as the measurements obtained from two networks N_1 and N_2. Therefore, the above argument applies.

3.3 FAULT PREDICTION APPROACH

As mentioned previously, in order to weaken the error effect due to component tolerance, the error term must be kept as small as possible. One way to solve this problem is to monitor the voltage measurements at the accessible terminals continuously at each periodic maintenance.

The voltage measurements monitored in two consecutively scheduled maintenances are assumed to be the voltages measured at accessible terminals of network N' and $\overline{N}'$. It is reasonable to assume that network N' is nonfaulty, since the fault prediction process for the previous maintenance has predicted no faulty component in N'. However, network $\overline{N}'$ may or may not be faulty. If $\overline{N}'$ is faulty, then networks N' and $\overline{N}'$ are referred to as the *prefault* and *postfault* network with tolerance, respectively.

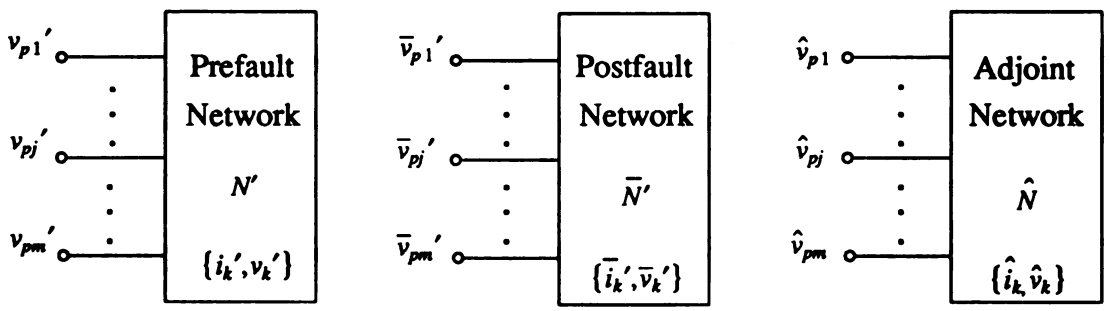


Figure 3.3 Prefault and Postfault Networks.

Consider networks N', $\overline{N}'$, and the adjoint network $\hat{N}$, as shown in Figure 3.3. Based on Tellegen theorem, network N' has

$$\sum_{k=1}^{b} [\hat{v}_k(i_k+\Delta i_k')-(v_k+\Delta v_k')\hat{i}_k] = -\sum_{j=1}^{m} [\hat{v}_{pj}(i_{pj}+\Delta i_{pj}')-(v_{pj}+\Delta v_{pj}')\hat{i}_{pj}] \tag{3.6}$$

Subtracting (3.6) from (3.1), we have

$$\sum_{k=1}^{b} [(\hat{v}_k\Delta\overline{i}_k'-\Delta\overline{v}_k'\hat{i}_k)-(\hat{v}_k\Delta i_k'-\Delta v_k'\hat{i}_k)] = -\sum_{j=1}^{m} [\hat{v}_{pj}(\overline{i}_{pj}'-i_{pj}')-(\overline{v}_{pj}'-v_{pj}')\hat{i}_{pj}] \tag{3.7}$$

With the branch constraints,

$$i_k' = y_k' v_k', \quad i_k+\Delta i_k' = (y_k+\Delta y_k')(v_k+\Delta v_k'),$$

$$\bar{i}_k' = \bar{y}_k' \bar{v}_k', \quad i_k+\Delta \bar{i}_k' = (y_k+\Delta \bar{y}_k')(v_k+\Delta \bar{v}_k'),$$

and

$$\Delta \bar{y}_k' = \Delta y_k' + \Delta y_k$$

$$LHS\ of\ (3.7) = \sum_{k=1}^{b} (\hat{v}_k \Delta \bar{y}_k' \bar{v}_k' - \hat{v}_k \Delta y_k' v_k')$$

$$= \sum_{k=1}^{b} [\hat{v}_k (\Delta y_k' + \Delta y_k) \bar{v}_k' - \hat{v}_k \Delta y_k' v_k']$$

$$= \sum_{k=1}^{b} (\hat{v}_k \Delta y_k \bar{v}_k) + \sum_{k=1}^{b} [\hat{v}_k \Delta y_k' (\bar{v}_k' - v_k') - \hat{v} \Delta y_k (\bar{v}_k' - \bar{v}_k)]$$

Therefore, Equation (3.7) can be written as

$$\sum_{k=1}^{b} \hat{v}_k \Delta x_k = \Delta p'' - \Delta e'' \tag{3.8}$$

where

$$\Delta x_k = \Delta y_k \bar{v}_k$$

$$\Delta p'' = \sum_{j=1}^{m} [(\bar{v}_{pj}' - v_{pj}') \hat{i}_{pj} - \hat{v}_{pj} (\bar{i}_{pj}' - i_{pj}')]$$

$$\Delta e'' = \sum_{k=1}^{b} [\hat{v}_k \Delta y_k' (\bar{v}_k' - v_k') + \hat{v}_k \Delta y_k (\bar{v}_k' - \bar{v}_k)]$$

Δy_k, $\Delta y_k'$, $\bar{v}_k$ and $\bar{v}_k'$ are the same as defined preciously, $(\bar{v}_k' - v_k')$ is the difference of the kth branch voltages of prefault and postfault networks, and $(\bar{v}_k' - \bar{v}_k)$ is the difference of the kth branch voltages of the faulty network with and without component tolerance.

Similar to the Δp of Equation (2.7), $\Delta p''$ is the difference of two voltage measurement at the same jth terminals of N' and $\bar{N}'$. Consider error term $\Delta e''$, the second term of $\Delta e''$ is the same as that of $\Delta e'$. Thus, this term has been shown to be negligible.

Furthermore, it is obvious that the deviation $(\bar{v}_k'-v_k')$ of $\Delta e''$ is much smaller than v_k' of $\Delta e'$ in the first term. In the case that components fail due to permanent overstress, the component parameters change sufficiently slow. The resulting deviation $(\bar{v}_k'-v_k')$ is so small that the error effect of $\Delta e''$ can be totally negligible.

The fault prediction algorithm is summarized as follows:

Algorithm II.

Step 1. Retrieve the previous measurements v_{pj}' from the database.

Step 2. Input the voltage $\bar{v}_{pj}'$ measured at the jth terminal.

Step 3. Check whether or not the system is functioning in a specific safety range.

Step 4. If so, GOTO Step 2. Otherwise, GOTO Step 5 to predict and locate the faulty components.

Step 5. Calculate $\Delta p_j'' = \bar{v}_{pj}'-v_{pj}'$, $j=1,2,\cdots,m$.

Step 6. Similar to the Step 3 in Algorithm I, locate and evaluate the component values.

Step 7. Check whether or not the components deviated within the prescribed tolerance.

Step 8. If so, GOTO Step 2. Otherwise, display the faulty components that have been predicted.

3.4 EXAMPLES

In order to demonstrate the effectiveness of the proposed fault prediction algorithm, two examples are given as follows.

Example 3.1

For the same network as shown in Figure 2.3, assume that component tolerance is within ±5%. Suppose that faulty components are r_2=0.5 and r_{18}=0.5. If we directly use the fault location algorithm shown in Section 3.1, the computed node voltages V and measured node voltages $\bar{V}'$ and $\Delta P'$ are listed in Table 3.1.

Table 3.1 Simulation Data 1 of Example 3.1

Node	V	$\bar{V}'$	$\Delta P'$
1	1.100000*e*+00	9.659305*e*−01	−1.340695*e*−01
6	5.500000*e*−01	4.638656*e*−01	−0.861344*e*−01
7	5.500001*e*−01	5.153373*e*−01	−0.346627*e*−01

Similar to the fault location process discussed in Example 2.1, with the use of the W-matrix in Table 2.3, we cannot find the fault set candidate that satisfies the condition $rank[W_f \mid \Delta P'] = rank[W_f] = f$, i.e., the faulty components cannot be located properly.

With the proposed fault prediction algorithm, the voltage measurements monitored in prefault and postfault network V' and $\bar{V}'$ are listed in Table 3.2.

Table 3.2 Simulation Data 2 of Example 3.1

Node	V'	$\bar{V}'$	$\Delta P''$
1	1.110968*e*+00	9.659305*e*−01	−1.450375*e*−01
6	5.670416*e*−01	4.638656*e*−01	−1.031760*e*−01
7	5.668063*e*−01	5.153373*e*−01	−0.514690*e*−01

With the same W-matrix of Table 2.3, we find the set {2,18} is the actual fault set, i.e., the components 2 and 18 are known as faulty, and their component values are estimated as r_2=0.494 and r_{18}=0.468, which are very close to the given simulated fault values.

Example 3.2

Consider an active filter whose component nominal values are shown in Figure 3.4. Assume component tolerance is within ±5%. Suppose that a voltage source of 0.5v at 10kHz is applied. Assume that faulty component R_8=1500Ω is simulated.

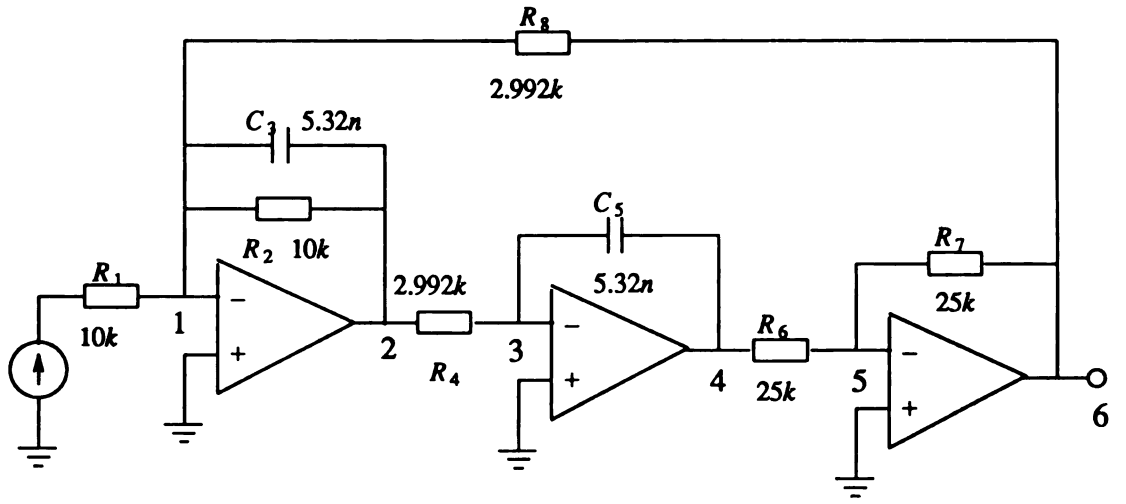


Figure 3.4 An Active Filter.

The voltage measurements monitored in prefault and postfault networks, V' and $\bar{V}'$, are listed in Table 3.3.

Table 3.3 Simulation Data of Example 3.2

Node	V'	$\bar{V}'$
1	(−5.486664*e*−04,−1.632575*e*−02)	(−1.354974*e*−04,−7.895228*e*−03)
6	(1.635527*e*−01,−5.407389*e*−03)	(7.909434*e*−02,−1.314312*e*−03)
7	(−1.558335*e*−01, 4.979997*e*−03)	(−7.536009*e*−02, 1.163744*e*−03

Following the fault prediction procedure, we can find that the set {8} is the actual fault set , i.e, the component 8 is known as faulty, and its component value is estimated as R_8=1487Ω, which is very close to the given simulated fault values.

3.5 DISCUSSION AND SUMMARY

In Chapters 2 and 3, fault location and fault prediction algorithms are presented, in which faults can be precisely located and potential faulty components can be predicted before the system actually fails.

In the fault location and prediction algorithms discussed in Algorithms I and II, the major computational requirement in the off-line phase are (1) to construct matrix H; (2) to calculate branch voltages $\hat{v}_j^{(i)}$ for adjoint networks to build matrix W; and (3) to calculate node voltages V for the fault-free network N. In the on-line phase, the computation includes the rank checking in Equation (2.10) for all possible matrices W_f's. When the size of the circuit network is reasonably small, the dimensions of matrices and equations will also be small. Consequently, the computation requirement for both on-line and off-line may not dominate the speed of diagnosis. However, when the size of the circuit is increased, both the dimensions of matrices and equations in the off-line phase and the component combinations for the rank checking in the on-line phase are increased. That results in the requirement of expensive off-line and on-line computations. Therefore, the proposed algorithm is not feasible to diagnose faults in large networks.

However, if a practical large network can be decomposed into several small subnetworks, then the proposed fault location and prediction algorithms can be applied to diagnose the subnetworks with a reasonably high speed. This motivates the development of a decomposition approach fault diagnosis algorithm for large circuit networks, which will be discussed in the next chapter.

CHAPTER 4

DECOMPOSITION APPROACH FAULT PREDICTION

With the rapidly increasing complexity and size of modern electronic systems, the ability to adequately design a diagnosable system is a prime requisite for rapid fault location. The fault location and prediction algorithms discussed in previous chapters can precisely locate and predict faulty components for small size networks. Due to the substantial increase of on-line computation for a large network, however, direct implementation of such algorithms for a practically large circuit network would be impractical.

In practice, a faulty network generally has failures in a small portion of the network and the remaining parts are fault-free. Decomposing a large circuit network into several small subnetworks seems a very attractive idea. Recently, a decomposition approach fault diagnosis process has been proposed [39]. The approach can precisely locate faults when component values assume their nominal. However, the error effect due to component tolerance in practical circuits may drastically affect the identification of the faulty components.

In this chapter, the decomposition approach originally proposed by [39] is discussed. The inherent problems in such an approach are also pointed out. In order to alleviate error effect, a decomposition approach fault prediction algorithm is presented to precisely predict faults in large circuit networks.

4.1 DECOMPOSITION APPROACH

Consider the original decomposition approach proposed by Salama, Starzyk, and Bandler [39]. In this approach, a nodal decomposition of a network into smaller uncoupled subnetworks is carried out; The measurement nodes (or, accessible nodes) must include the nodes of decomposition (for simplicity, these nodes are referred to as D-nodes); The voltage measurements are employed to isolate the faulty subnetworks; The incidence current relation between subnetworks are checked against the KCL (Kirchhoff's Current Law) to determine the status of those D-nodes; And a logic analysis of the results is carried out to identify faulty subnetworks.

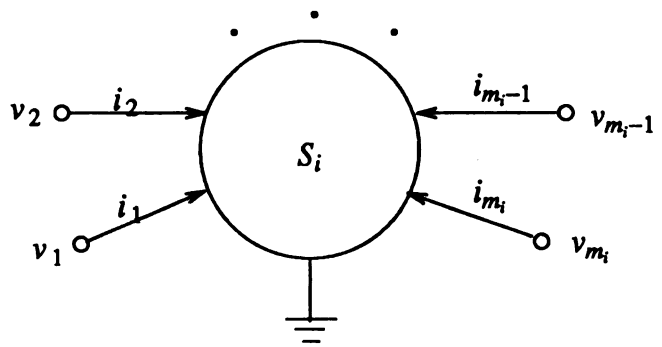


Figure 4.1 A Subnetwork S_i with m_i External Nodes.

Consider a linear subnetwork S_i as shown in Figure 4.1. It consists of n_i internal nodes and connects to other subnetworks through the m_i external nodes. Let V^{n_i} and V^{m_i} be the voltage vectors of the n_i internal nodes and m_i external nodes, respectively, and I^{m_i} be the incidence current vector of the subnetwork S_i from the m_i external nodes, the nodal equations of the subnetwork S_i then are expressed as follows

$$\begin{bmatrix} Y_{m_i m_i} & Y_{m_i n_i} \\ Y_{n_i m_i} & Y_{n_i n_i} \end{bmatrix} \begin{bmatrix} V^{m_i} \\ V^{n_i} \end{bmatrix} = \begin{bmatrix} I^{m_i} + I_g^{m_i} \\ I_g^{n_i} \end{bmatrix} \tag{4.1}$$

where

$$I_g^i = \begin{bmatrix} I_g^{m_i} \\ I_g^{n_i} \end{bmatrix}$$

is the current sources associated with the subnetwork.

Eliminating the n_i internal nodes yields [39]

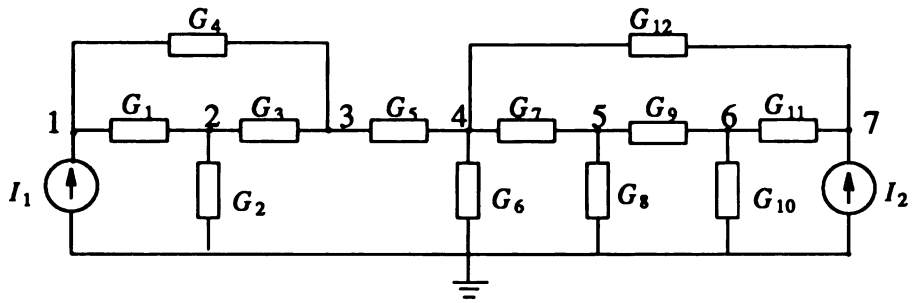
$$I^{m_i} = -[I_g^{m_i} - Y_{m_i n_i}\, Y_{n_i n_i}^{-1}\, I_g^{n_i}] + [Y_{m_i m_i} - Y_{m_i n_i}\, Y_{n_i n_i}^{-1}\, Y_{n_i m_i}]\, V^{m_i} \tag{4.2}$$

The sum of the computed currents associated with each D-node is checked against the KCL. If the computed currents satisfy the KCL at a D-node, all subnetworks connected to this node are fault-free. Otherwise, at least one of these subnetworks is faulty. In fact, if the currents associated with a fault-free node satisfy the KCL, the sum of such currents should be zero. In practice, however, a reasonably small tolerant term ε is used as the bounds. In other words, if the sum of the currents is less than a predefined tolerant term ε, then the node is fault-free; otherwise the node is faulty.

In order to demonstrate the decomposition approach proposed in [39], two examples are discussed. Example 4.1 will show that the approach can identify the faults precisely when the component values assume their nominals. Followed by Example 4.2 which illustrates the approach for the case that each component deviates from its nominal within 5%.

Example 4.1.

Consider a linear network as shown in Figure 4.2 [10], the component values assume their nominals exactly.



G_1	1.5	G_7	3.0
G_2	1.8	G_8	1.3
G_3	2.0	G_9	2.3
G_4	0.5	G_{10}	3.1
G_5	2.6	G_{11}	2.9
G_6	1.2	G_{12}	0.8

Figure 4.2 A Linear Network.

Suppose that component 7 is faulty and its component value is changed from $G_7 = 3.0$ to 2.0. We first decomposed the network of Figure 4.2 into S_1 and S_2 as shown in Figure 4.3, where nodes 1, 4, and 7 are selected as D-nodes.

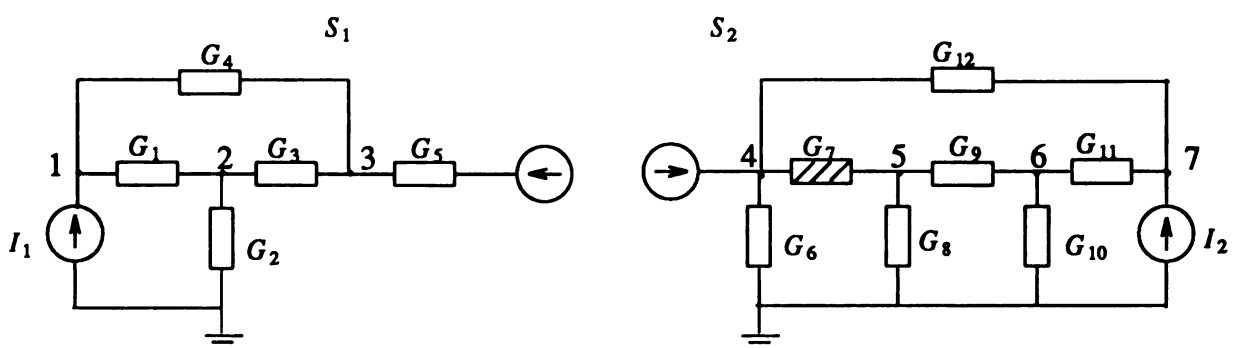


Figure 4.3 Decomposed Subnetworks S_1 and S_2.

Table 4.1 shows the simulation results of the first level of decomposition, where I_j^i represents the incidence current of the subnetwork S_i from the D-node j, and $\sum I^{m_i}$, Column 6 of Table 4.1, denotes the sum of the computed currents at the D-node j. Column 6 shows that the summation of currents at the node 1 is approximately zero, and

concludes that the node 1 is fault-free. Since the current summations at both nodes 4 and 7 are far away from zero, both nodes are determined as faulty. Therefore, the subnetwork S_1 is determined as fault-free, but S_2 is as faulty. From Figure 4.3, the faulty component is indeed contained in S_2.

Table 4.1

Node j	V^{m_i}	I_g	Subnetwork 1	Subnetwork 2	$\sum I^{m_i}$
1	1.721510	2.00000	I_1^1= 2.00000	–	-5.551115e-17
4	0.410680	0.00000	I_4^1=-0.66922	I_4^2=0.74504	-7.582164e-2
7	0.571673	1.00000	–	I_7^2=0.97943	2.056651e-2

The faulty subnetwork may be decomposed further if necessary. Figure 4.4 shows that the faulty subnetwork S_2 is further decomposed into S_3 and S_4, where the nodes 4, 6, and 7 are selected as D-nodes, and the current source I_{S_1} is obtained from the previous level of decomposition, i.e., $I_{S_1} = -I_4^1$ =0.66922 (Table 4.1).

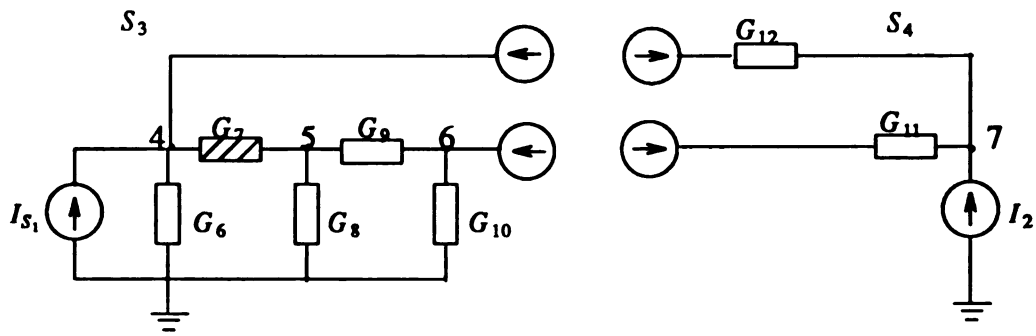


Figure 4.4 Decomposed Subnetworks S_3 and S_4.

Table 4.2 lists the simulation results and concludes that subnetwork S_4 is fault-free, but S_3 is faulty.

Table 4.2

Node j	V^{m_i}	I_g	Subnetwork 3	Subnetwork 4	$\sum I^{m_i}$
4	0.410680	0.66922	I_4^3=0.88125	I_4^4=0.12879	-8.323550e-2
6	0.271258	0.00000	I_6^3=0.81803	I_6^4=-0.87121	5.317849e-2
7	0.571673	1.00000	–	I_7^4= 1.00000	5.551115e-17

Example 4.2.

Consider the same network of Figure 4.2, but with component values deviating from their nominal values within a random tolerance of 5%. Suppose also that the component 7 is faulty. Similar to the procedures discussed in Example 4.1, Table 4.3 lists the simulation results of the first level of decomposition that decomposes the network into S_1 and S_2 with the same external nodes.

Table 4.3

Node j	V^{m_i}	I_g	Subnetwork 1	Subnetwork 2	$\sum I^{m_i}$
1	1.716525	2.00000	I_1^1= 1.9894	–	1.064765e-2
4	0.417103	0.00000	I_4^1=0.65933	I_4^2=0.77128	-1.119470e-1
7	0.568505	1.00000	–	I_7^2=0.96353	3.647195e-2

With an predefined tolerant term ε, it is still possible to conclude that both nodes 4 and 7 are faulty, but node 1 is fault-free. Consequently, the subnetwork S_1 is determined as fault-free, but S_2 is as faulty.

For the next level of decomposition, Table 4.4 lists the simulation results.

Table 4.4

Node j	V^{m_i}	I_g	Subnetwork 3	Subnetwork 4	$\sum I^{m_i}$
4	0.417013	0.65933	I_4^3=0.89364	I_4^4=-0.12112	-1.131928e-1
6	0.276830	0.00000	I_6^3=0.83693	I_6^4=-0.84586	8.922686e-3
7	0.568505	1.00000	–	I_7^4= 0.96698	3.302115e-2

If we denote τ_k as the sum of computed currents associated with the D-node k, Column 6 of Table 4.4 shows that $\tau_4 > \tau_7 > \tau_6$. We claim that it is impossible to find any tolerant term ε to precisely identify the faulty component 7 in this approach. Two cases can be identified for the range of the ε: either it lies between τ_6 and τ_7, or between τ_4 and τ_7. If ε lies between τ_6 and τ_7, then both nodes 4 and 7 are determined as faulty, but node 6 is fault-free. Further, since all the subnetworks connected to a fault-free node are fault-free and node 6 is fault-free, both subnetworks S_3 and S_4 are thus fault-free. This implies that both nodes 4 and 7 must be fault-free, which contradicts the above determination.

On the other hand, if we assume that ε lies between τ_4 and τ_7. This yields that nodes 6 and 7 are fault-free, but node 4 is faulty. Similarly, subnetwork S_3 is concluded as fault-free and results that node 4 must be fault-free which is also a contradiction. Therefore, it is impossible to predefine a tolerant term to determine the faulty component in this approach.

Example 4.2 shows that the decomposition approach in [39] may not precisely identify the faulty components when the component tolerance is taken into account. The fact is that an error effect due to the component tolerance could affect the fault identification process. In our observation, the major problem of that decomposition approach is that the error effect is induced not only by the component tolerance, but also by the accumulation of the computation error. More specifically, in the second level of

decomposition, for example, the current source I_{S_1}, as shown in Figure 4.4, is obtained from the results computed in the first level of decomposition. In Example 4.2, $I_{S_1} = -I_4^1 = 0.65933$, which differs from the current $I_{S_1} = -I_4^1 = 0.66922$ obtained in Example 4.1. This computed error is carried forward for the computation for the next level of decomposition. As a result, as more levels of decomposition are executed, the higher the induced error accumulation may be. In fact, the error effect can be alleviated if the computed current is avoided. This motivates the development of a decomposition approach fault prediction algorithm that is presented in next section.

4.2 FAULT PREDICTION APPROACH

Consider Equation (4.1), if we reorder the equation as

$$\begin{bmatrix} Y_{n_i n_i} & Y_{n_i m_i} \\ Y_{m_i n_i} & Y_{m_i m_i} \end{bmatrix} \begin{bmatrix} V^{n_i} \\ V^{m_i} \end{bmatrix} = \begin{bmatrix} I_g^{n_i} \\ I^{m_i} + I_g^{m_i} \end{bmatrix} \tag{4.3}$$

and apply the forward-elimination steps of the Gaussian elimination method to Equation (4.3), then we obtain

$$\begin{bmatrix} U & Y_{n_i m_i}{}' \\ 0 & Y_{m_i m_i}{}' \end{bmatrix} \begin{bmatrix} V^{n_i} \\ V^{m_i} \end{bmatrix} = \begin{bmatrix} I_g^{n_i\prime} \\ I^{m_i} + I_g^{m_i\prime} \end{bmatrix} \tag{4.4}$$

where U is an $n_i \times n_i$ upper triangular matrix with all 1's in the major diagonal, and 0 is an $m_i \times n_i$ zero matrix. Equation (4.4) gives the following nodal equation of m_i's external nodes

$$Y_{m_i m_i}{}' V^{m_i} = I_g^{m_i\prime} + I^{m_i} \tag{4.5}$$

or

$$I^{m_i} = -I_g^{m_i} + Y_{m_i m_i}{}' V^{m_i} \tag{4.6}$$

where $Y_{m_i m_i}{}'$ can be computed off-line using component nominal values. Denote vectors $V_1^{m_i}$ and $V_2^{m_i}$ as the voltages at external nodes of a subnetwork S_i measured at two

consecutive cycles. From Equation (4.6), we have

$$Y_{m_i m_i}{}' V_1^{m_i} = I_g^{m_i}{}' + I_1^{m_i} \tag{4.7}$$

$$Y_{m_i m_i}{}' V_2^{m_i} = I_g^{m_i}{}' + I_2^{m_i} \tag{4.8}$$

Subtracting Equation (4.7) from (4.8)

$$\Delta I_{m_i} = I_2^{m_i} - I_1^{m_i} = Y_{m_i m_i}{}' (V_2^{m_i} - V_1^{m_i}) = Y_{m_i m_i}{}' \Delta V_{m_i} \tag{4.9}$$

It should be pointed out that the computation of Equation (4.6) is much simpler than that of Equation (4.2). This gains a significant improvement in computation. Also, Equation (4.9) shows that the external currents $I_g^{m_i}$ are eliminated and thus the error effect can be alleviated significantly.

Based on the above concept, the decomposition approach fault prediction algorithm is summarized in Algorithm III.

Algorithm III

Step 1. Decompose the network at all the accessible (external) nodes (by logically breaking the connection at the nodes) to obtain smallest possible subnetworks.

Step 2. Compute $Y_{m_i m_i}{}'$ by using the nominal parameter values of each subnetwork S_i.

Step 3. Retrieve the previous measurement V_1^m from the data base.

Step 4. Input the Voltage V_2^m measured at decomposition nodes.

Step 5. Calculate $\Delta V_m = V_2^m - V_1^m$ and check whether the system is properly functioning in a specified safety range.

Step 6. If so, (no fault is detected) GOTO Step 4.

Otherwise, (to identify the faulty subnetworks).

Step 7. $\Delta I_{m_i} = Y_{m_i m_i}{}' \Delta V_{m_i}$

Step 8. If the computed currents satisfy KCL at a D-node, all the subnetworks connected to this node are fault-free. Otherwise at least one of these subnetworks is faulty.

In order to demonstrate the effectiveness of the proposed algorithm, Example 4.3 is discussed below.

Example 4.3.

Consider the same network and same conditions as in Example 4.2. Following Algorithm III, Tables 4.5 and 4.6 list the simulation results for two levels of decomposition illustrated in Table 4.3 and 4.4, respectively.

Table 4.5

Node	$V_1^{m_i}$	$V_2^{m_i}$	ΔV_{m_i}	$\sum \Delta I_{m_i}$
1	1.706806	1.716525	9.718500e-3	-2.112174e-4
4	0.397415	0.417103	1.968805e-2	-8.283782e-2
7	0.568300	0.568505	2.045350e-4	2.318341e-2

Table 4.6

Node	$V_1^{m_i}$	$V_2^{m_i}$	ΔV_{m_i}	$\sum \Delta I_{m_i}$
4	0.397415	0.417103	1.968805e-2	-7.657193e-2
6	0.281749	0.276830	-4.919049e-3	5.806152e-2
7	0.568300	0.568505	2.045350e-4	7.284202e-4

In contrast to Example 4.2, it is possible to consistently select a tolerant term that can precisely determine the faulty component 7 from both Tables 4.5 and 4.6.

Example 4.4

Consider a signal filter circuit shown in Figure 4.5. Its component values are listed in Table 4.7. The network is decomposed into six subnetworks, as shown in Figure 4.6, where decomposition nodes are 1, 3, 5, 11, 13, 15, 21, and 22.

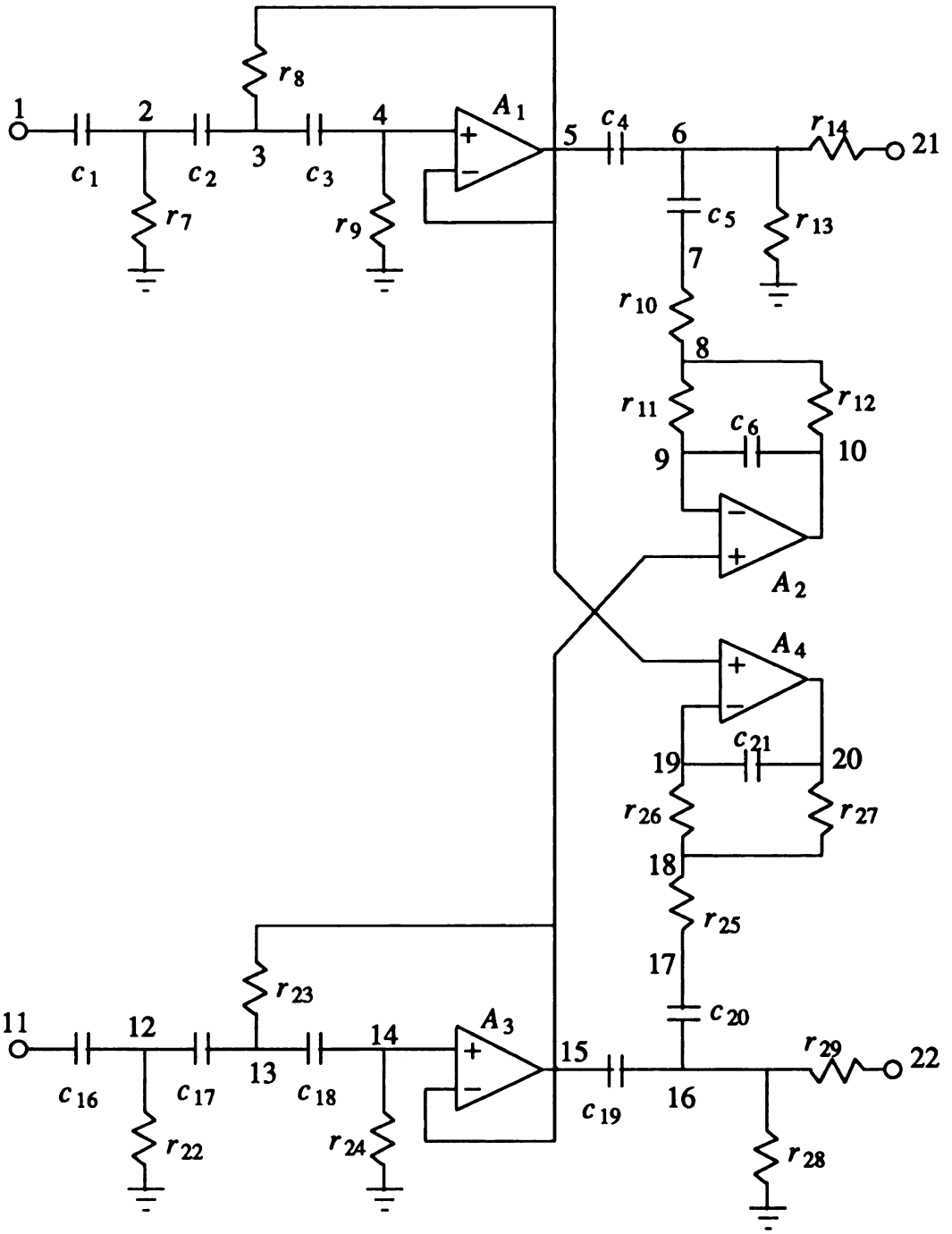


Figure 4.5 A Signal Filter Circuit.

Table 4.7 Component Values of Example 4.4

Component	value	Component	value
c_1, c_{16}	0.1μF	r_8, r_{23}	15kΩ
c_2, c_{17}	0.1μF	r_9, r_{24}	750kΩ
c_3, c_{18}	0.1μF	r_{10}, r_{25}	220Ω
c_4, c_{19}	2.2μF	r_{11}, r_{26}	39kΩ
c_5, c_{20}	2.2μF	r_{12}, r_{27}	1kΩ
c_6, c_{21}	0.01μF	r_{13}, r_{28}	56kΩ
r_7, r_{22}	56kΩ	r_{14}, r_{29}	560Ω

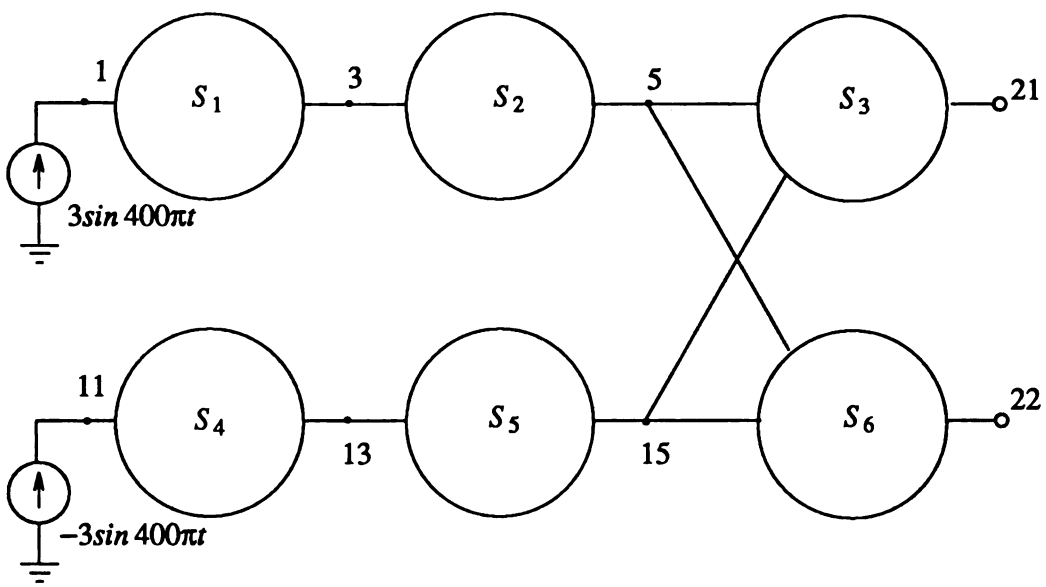


Figure 4.6 Decomposed Subnetworks.

Assume that r_9 is simulated as faulty component with r_9=7.5*k*. A transient analysis is performed with a 1*ms* time step. The simulation results are listed in Table

4.8, and the simulated results at t=10ms are summarized in Table 4.9.

Table 4.8 Simulation Result of Example 4.4

Time	node 1	node 3	node 5	node 21
0.000E+00	0. e+00	0. e+00	0. e+00	0. e+00
1.000e-03	2.849e-18	3.532e-05	6.372e+01	2.787e-21
2.000e-03	1.761e-18	-9.680e-05	2.716e+01	3.892e-20
3.000e-03	-1.760e-18	-8.152e-05	-7.678e+01	-1.028e-19
4.000e-03	-2.849e-18	5.648e-05	-9.200e+01	0.000E+00
5.000e-03	-3.463e-16	1.223e-04	1.035e+01	2.787e-21
6.000e-03	2.849e-18	2.249e-05	9.335e+01	-1.028e-19
7.000e-03	1.761e-18	-1.065e-04	4.490e+01	4.797e-20
8.000e-03	-1.760e-18	-8.720e-05	-6.657e+01	9.727e-20
9.000e-03	-2.849e-18	5.318e-05	-8.619e+01	-1.042e-19
1.000e-02	3.139e-18	1.205e-04	1.364e+01	-1.084e-19
Time	node 11	node 13	node 15	node 22
0.000E+00	0. e+00	0. e+00	0. e+00	0. e+00
1.000e-03	-2.849e-18	6.018e-20	-1.109e-14	2.432e-20
2.000e-03	-1.761e-18	2.662e-21	3.158e-14	-5.142e-20
3.000e-03	1.760e-18	4.000e-21	3.477e-14	-2.545e-20
4.000e-03	2.849e-18	-3.497e-20	2.411e-14	2.787e-21
5.000e-03	3.463e-16	-5.399e-21	3.687e-14	-2.787e-21
6.000e-03	-2.849e-18	2.908e-20	3.551e-14	2.571e-20
7.000e-03	-1.761e-18	1.102e-20	2.934e-14	4.864e-20
8.000e-03	1.760e-18	1.798e-20	3.257e-14	2.467e-20
9.000e-03	2.849e-18	2.019e-21	2.746e-14	-5.700e-20
1.000e-02	-3.139e-18	-5.880e-20	5.981e-14	-5.421e-20

Table 4.9 Analysis in One Time Step

node	ΔV_m	$\sum \Delta I$	subnetworks incident at node
1	−1.365e−04	3.139−18	S_1
3	6.727e−01	1.205−04	S_1, S_2
5	5.080e−01	1.364+01	S_2, S_3, S_6
21	−1.352e+00	−1.084e−19	S_3
11	−2.514e−09	−3.139e−18	S_4
13	4.966e−05	−5.880e−20	S_4, S_5
15	1.905e−05	5.981e−14	S_3, S_5, S_6
22	1.845e+00	−5.421e−20	S_6

Table 4.9 shows that the currents associated with the decomposition nodes 1, 11, 13, 15, 21 and 22 satisfy the KCL, and the currents at nodes 3 and 5 do not satisfy the KCL. This implies that the nodes in the former group are fault-free, and nodes 3 and 5 are faulty. As a result, the subnetwork S_2 connected to both faulty nodes 3 and 5 is known as faulty. As shown in Figure 4.5, the faulty component r_9 is, in fact, contained in this subnetwork.

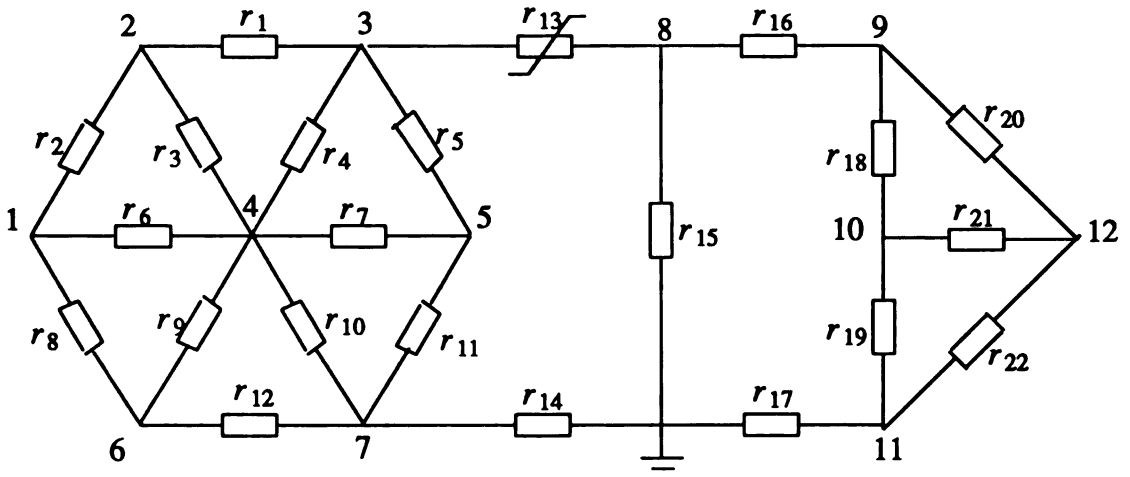
4.3 EXTENSION TO NONLINEAR CIRCUIT NETWORKS

Typically, the nonlinear network is dominantly linear with a few nonlinear components [39]. In general, the nonlinear components are "linearized" in order to be modeled by a digital computer. After linearization, even the most complex models contain only two different types of circuit components [9]: a resistor (impedance) and a current source, which may be voltage-dependent. Therefore, the decomposition approach developed for linear circuit networks can be applied for nonlinear networks. Basically, the nodes of decomposition are chosen such that the network is decomposed into blocks that contain the nonlinear components and a number of subnetworks that

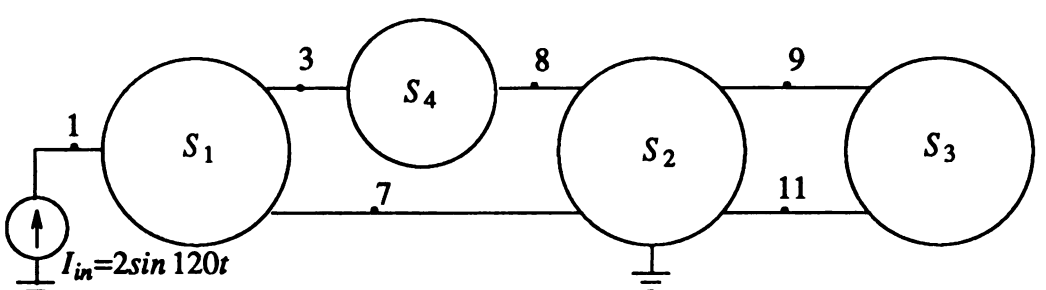
contain only linear components. In order to analyze the nonlinear subnetwork we use the SPICE program [30] as the network solver. A set of voltage controlled voltage sources are added to all decomposition nodes of the original circuit to accept the voltages measured at two consecutive cycles. The transient analysis is used to calculate the sum of currents flowing into the decomposition nodes at each time step.

Example 4.5

Consider a nonlinear network as shown in Figure 4.7(a), which consists of all linear resistors except component 13, where the nominal values of the linear resistor are 1Ω and the nonlinear resistor with the function $i=2v^3$. Suppose that components deviate from their nominal values within a random tolerance of 5%. The network is first decomposed into four subnetworks S_1, S_2, S_3, and S_4, as shown in Figure 4.7 (b), where S_4 only consists of the nonlinear component r_{13}, and decomposition nodes are 1, 3, 7, 8, 9, and 11.



(a)



(b)

Figure 4.7 (a) A Network (b) Decomposed Subnetworks.

Two cases are simulated in this example: (1) the nonlinear component r_{13} is faulty; and (2) two linear components r_1 and r_9 are faulty.

In the former case, the i-v function of the component r_{13} is assumed to be changed to $i=4v^3+1$. Following Algorithm III, the voltages measured at the decomposition nodes and the sum of computed currents associated with each decomposition node at one time step are listed in Table 4.10.

Table 4.10 Simulation Result 1 of Example 4.5

node	ΔV_m	$\sum \Delta I$	subnetworks incident at node
1	1.434e−01	−4.163e−06	S_1
3	1.825e−01	−1.396e−01	S_1, S_4
7	1.044e−01	−2.048e−04	S_1, S_2
8	−7.825e−02	1.398e−01	S_2, S_4
9	−5.215e−02	−6.801e−05	S_2, S_3
11	−2.611e−02	6.801e−05	S_2, S_3

Table 4.10 shows that the currents associated with the decomposition nodes 1, 7, 9, and 11 satisfy the KCL, and the currents at both nodes 3 and 8 do not satisfy the KCL, which are thus claimed as the faulty nodes. As a result, subnetwork S_4 connected to these two nodes is found as faulty. In fact, the faulty nonlinear component r_{13} is in this subnetwork.

In the latter case, linear components with $r_1=5\Omega$ and $r_9=10\Omega$ are simulated as faulty. Following Algorithm III, the voltages measured at the decomposition nodes and the sum of computed currents associated with each decomposition node at one time step are listed in Table 4.11.

Table 4.11 Simulation Result 2 of Example 4.5

node	ΔV_m	$\sum \Delta I$	subnetworks incident at node
1	−1.563*e*−01	2.780*e*−01	S_1
3	6.452*e*−02	−3.073*e*−01	S_1, S_4
7	−6.034*e*−02	8.601*e*−02	S_1, S_2
8	4.525*e*−02	−6.033*e*−05	S_2, S_4
9	3.015*e*−02	3.932*e*−05	S_2, S_3
11	1.510*e*−02	−3.932*e*−05	S_2, S_3

Table 4.11 shows that the currents associated with the decomposition nodes 8, 9, and 11 satisfy the KCL, and the currents at nodes 1, 3 and 7 do not satisfy the KCL, which are thus claimed as the faulty nodes. As a result, the subnetwork S_1 that connects to these three nodes is found as faulty. As shown in Figure 4.7, the faulty components r_1 and r_9 are, in fact, contained in this subnetwork.

CHAPTER 5

BIST STRUCTURE

As discussed previously, faults can be precisely located at any desired level if sufficiently many test points are provided. Unfortunately, modern electronic systems are often multi-layered and/or coated, thereby limiting the accessibility of test points which are available at the externally accessible terminals of a printed circuit board. As the number of components in a unit increases, it is difficult to provide proportionately more I/O terminals.

The built-in self-test (BIST) design [27, 28] has been commonly implemented for digital circuit testing. The BIST design virtually increases the number of test points while still keeping low pin overhead. Circuits that generate test patterns and analyze the output responses of the functional circuitry are included on the same chip or elsewhere on the same board.

In this chapter, the concept of BIST design developed for digital circuits is adopted to the testing of analog circuit networks. An analog BIST structure is proposed and its simulation and hardware implementation are also presented.

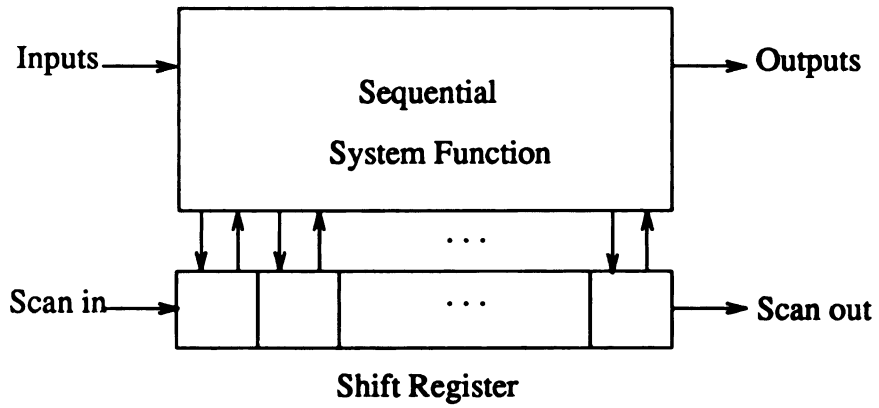
5.1 BIST STRUCTURES

In digital circuit, most BIST designs use some scan-path techniques, as shown as Figure 5.1(a), with shift register latchs (SRL). Figure 5.1 [4] shows that SRL contains two latches, L1 and L2. L1 constitutes the normal stored-state holding device with

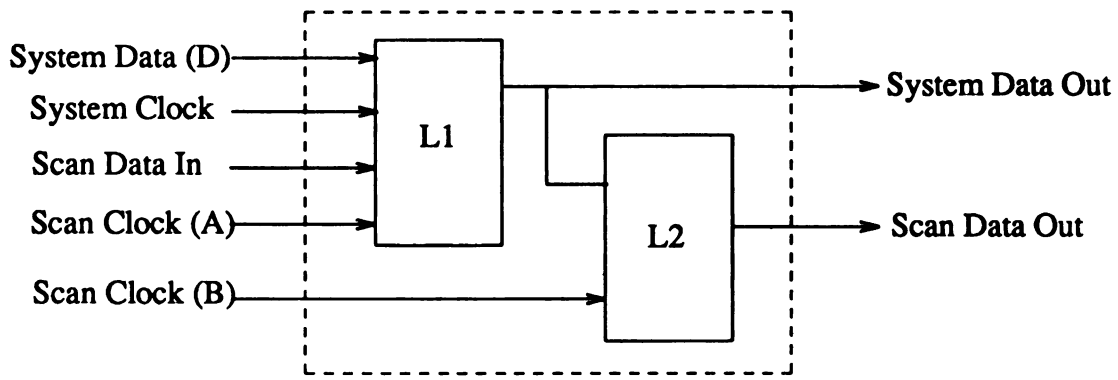
system date (D) and system clock input and system data output. In normal operation, scan clocks A and B are both held low. Latching of system data then occurs as the system clock is returned low from an active high value.

To operate the SRL as part of a scan path, scan clock A is set to 1. This enables data on the Scan-in to be latched directly into latch L1. Scan clock A is then returned low (to latch the value into L1) and scan clock B raised high. This causes transfer of the latched L1 value into latch L2 with permanent latching in L2 as B is returned low. The SRLs are connected to form a scan-path shift register by connecting the L2 output of one SRL to the input of another SRL. The two scan clocks, A and B, are common to all SRLs. A shift register with a chip containing 3 *SRL*'s is shown in Figure 5.1 (c)

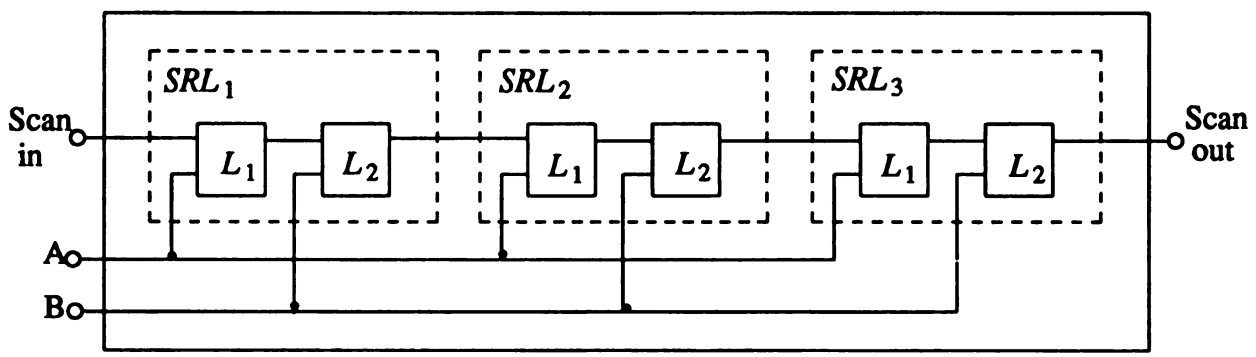
By adopting a design concept similar to a digital shift register, one may intuitively design an analog shift register (ASR), as shown in Figure 5.2 (a). The latch of the SRL in Figure 5.1 (b) can be realized by an sample-and-hold circuit. A BIST structure with analog shift registers is illustrated in Figure 5.2(b). Each unit (corresponding to a test point) requires a test point buffer B_i and an analog shift register ASR_i, where each analog shift consists of two simple sample-and-hold (S/H) circuits. (Each S/H circuit contains an analog switch, a holding capacitor, and a voltage follower.) The test point buffer B_i is used to isolate the test circuit from the UUT, so the test circuit will not affect the UUT, thus we can acquire voltage at the test point more accurately. In practice, a voltage follower having a very high input impedance and a low output impedance is employed. The buffer can then hold the test data without affecting the voltage level of the system during data sampling periods.



(a)

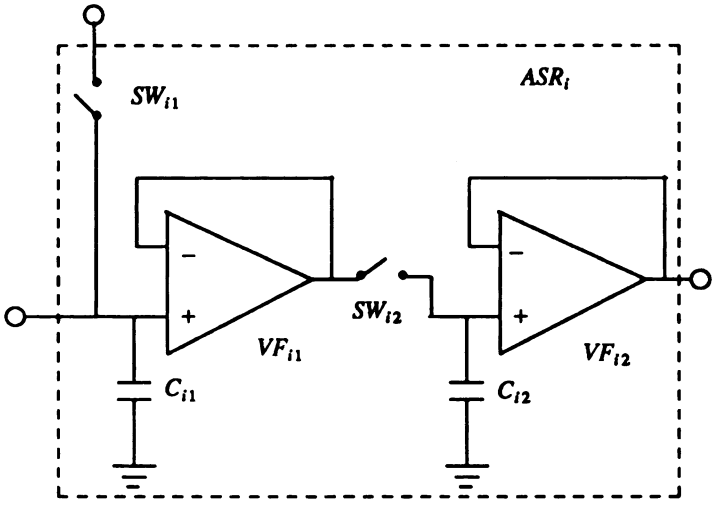


(b)

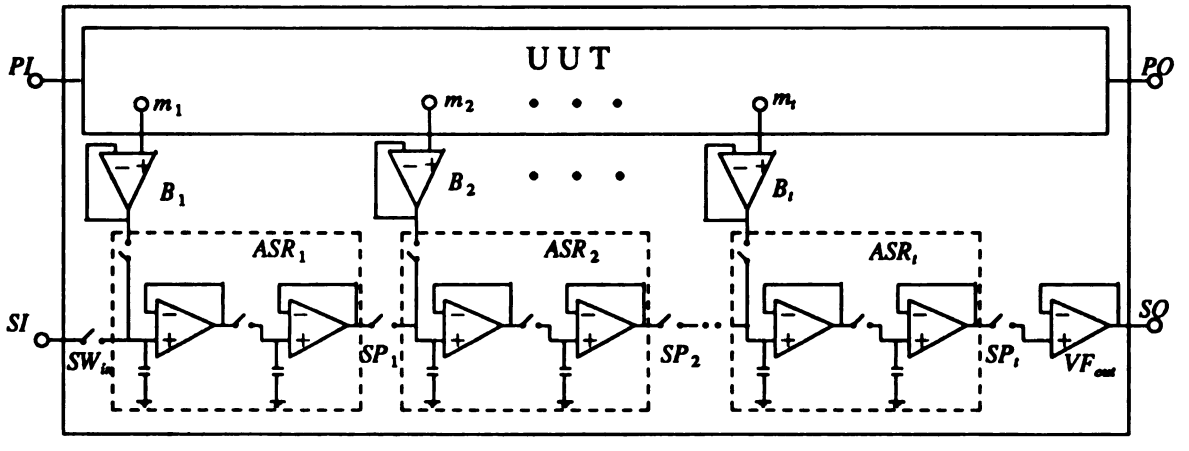


(c)

Figure 5.1 BIST Design of Digital Circuit:
(a) Scan Design Using Shift Register; (b) SRL;
and (c) a Shift Register with 3 SRLs.



(a)



(b)

Figure 5.2 BIST Structure with Analog Shift Register:
(a) Analog Shift Register ASR_i; and (b) Schematic Diagram.

This BIST structure allows the test data to be parallel sampled and serial shifted out for analog circuit testing, so the test data at various test points can be acquired simultaneously. The structure also virtually increases the number of test points while still keeping low pin overhead for analog circuit testing. However, each unit ASR_i requires two sample-and-hold circuits. Such design could increase chip area and decrease the operational speed. This motivates the following modification.

5.2 Analog BIST (ABIST) Structure

Figure 5.3 illustrates an analog BIST (ABIST) structure. Each unit (corresponding to a test point) of a ABIST structure requires a test point buffer B_i, an analog pass buffer APB_i, and a pass switch SP_i. Each analog pass buffer contains only one sample-and-hold circuit.

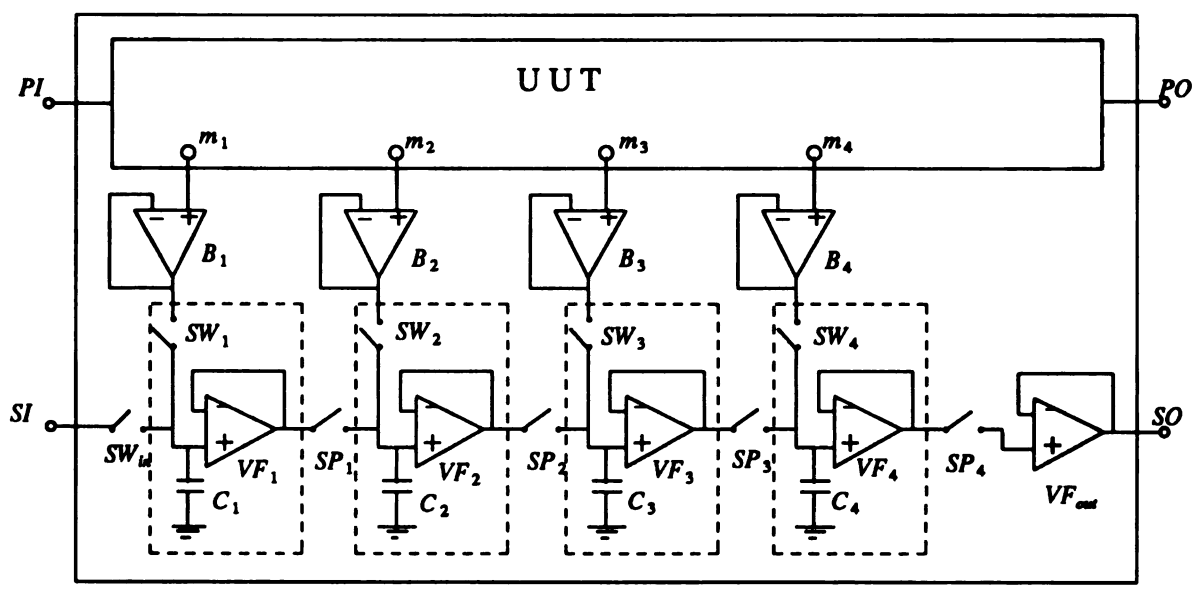


Figure 5.3 An Analog BIST Structure (ABIST).

A ABIST structure operates in two modes: normal operation mode and test mode. In the normal operation mode, all switches SW_{i1}'s are turned off and no test data are sampled. In the test mode, the test data are obtained in a fashion of "*parallel load and serial pass*".

During the "*loading*" period, all switches SW_i are turned on as shown in Figure 5.4 (a), thus the test data at the various test points are simultaneously loaded to the corresponding holding capacitors C_i's. This is followed by the "serial passing" process.

For simplicity of discussion, four test points are assumed in the UUT. First, switch SP_4 is turned on and remained at the on-state in the entire process, as shown in Figure 5.4(b). The test datum held on holding capacitor C_4 is passed to the terminal SO through both voltage follower VF_4 and the output buffer VF_{out}. Secondly, switch SP_3 is turned on, as shown in Figure 5.4(c), the test datum held on holding capacitor C_3 is then passed to the output. The subsequent test data are processed in the same way. The operation is carried out continually until the test datum held in holding capacitor C_1 is finally passed. The serial passing process is completed. The order of output sequence is "first stage last out".

Once the "serial passing" process is completed, the entire "loading and passing" process can then be carried out again.

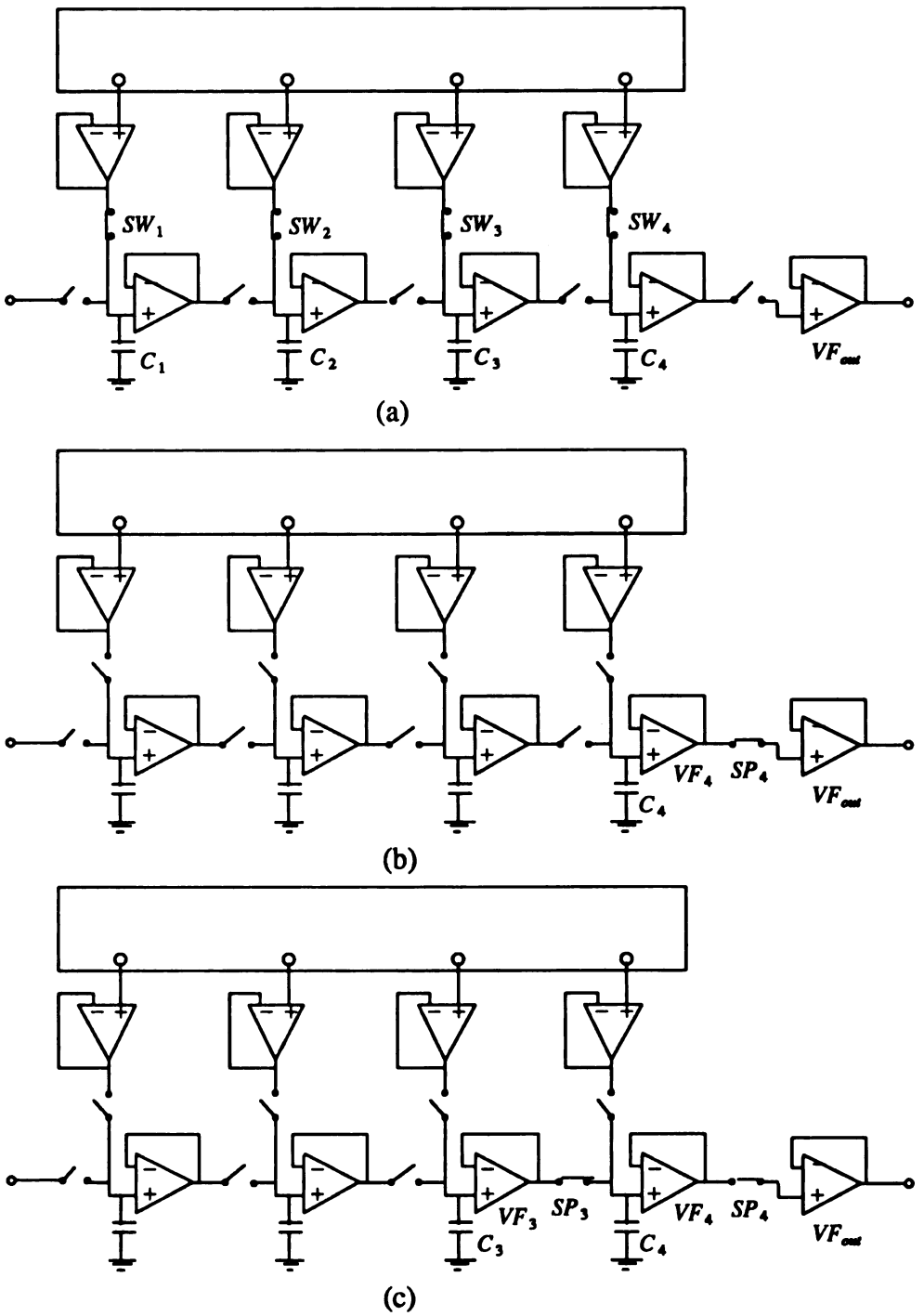


Figure 5.4 Switching Operation of the ABIST Structure:
(a) Parallel Loading; (b) & (c) Serial Passing.

Figure 5.5 shows the timing diagram of the switching operation, where clock signal ϕ_0 controls all switches SW_i's and clock signal ϕ_i controls corresponding switch SP_i, for i=1,2, $\cdots$,4. respectively.

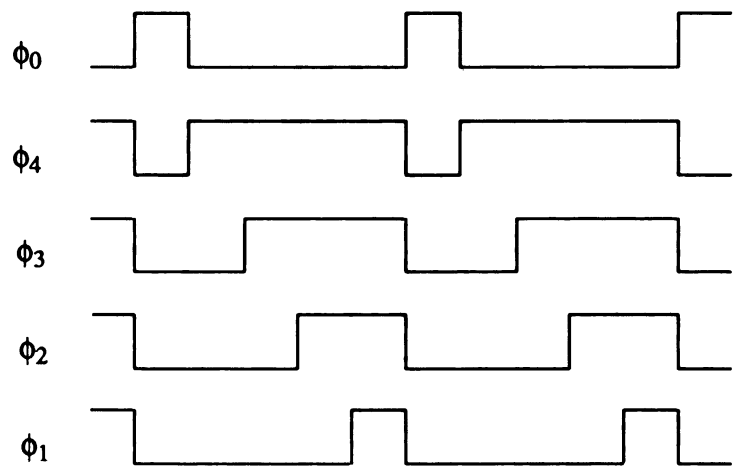


Figure 5.5 Timing Diagram of ABIST Structure.

The clock signals can be generated by the circuit of Figure 5.6, where each stage requires a D flip-flop and a 2-input OR gate.

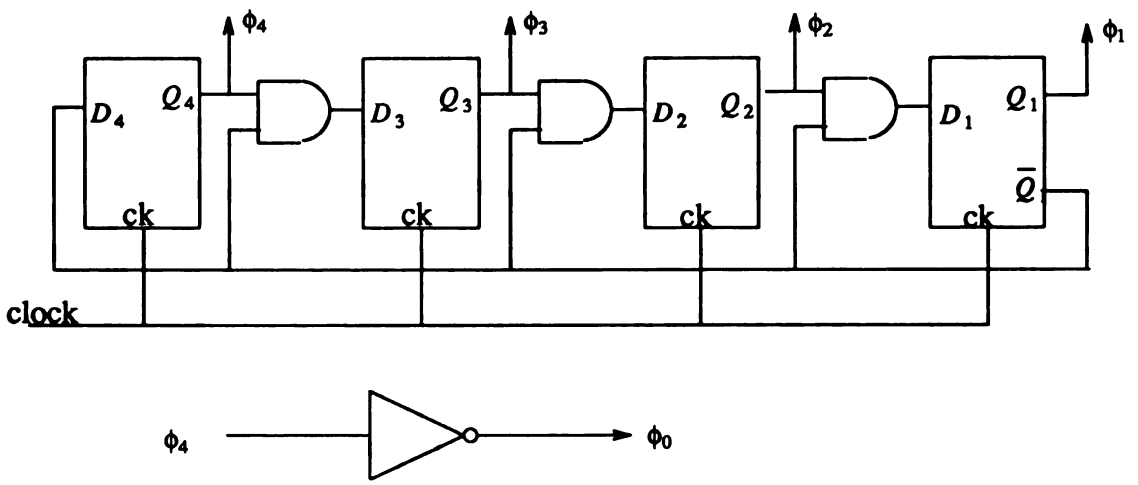


Figure 5.6 Clock Circuit.

It should be noted that the ABIST is self-testable by using the scan method. The extra pin overhead of the ABIST structure is the two pins, SI and SO. During the scan test process, switch SW_{IN} is turned on so that the test input signals can be applied to the passing buffers through the terminal SI (Scan-In). From output signals observed at terminal SO (Scan-Out), any mismatched signals indicate that faults occur in the passing buffers.

5.3 SIMULATION AND HARDWARE IMPLEMENTATION

The circuit simulation software package PSPICE is used to simulate the ABIST structure. For simplicity, a unity-gain op-amp model is used to simulate the voltage follower and the test point buffer. A PSPICE analog switch model is used to simulate the analog switches. The PSPICE input data for both DC and AC cases are shown in

Appendix 2, and the outputs are plotted in Figure 5.7.

A prototype of a 6-stage ABIST structure using buffer amplifier HA-5002 [11] and CMOS analog switch DG201A [41] has been implemented. In order to observe the output waveforms, an oscilloscope is connected to pin *SO* of the ABIST structure. Figure 5.8 displays the output waveforms of the experiments for both AC and DC analyses.

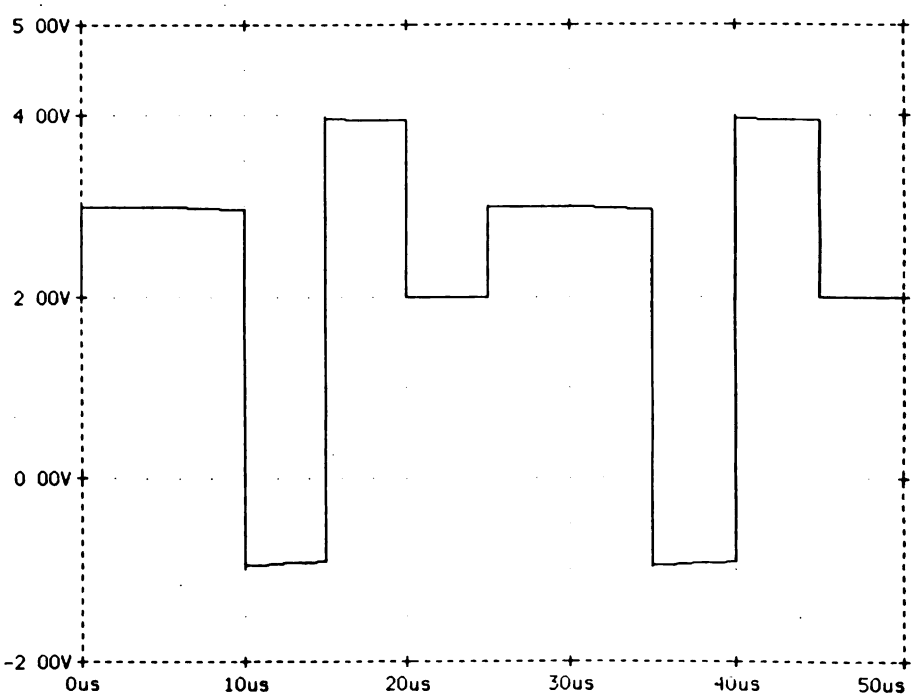
In our implementation, the output of the ABIST structure is connected to an A/D converter to record the test data for fault diagnosis use. However, the salient feature of the ABIST structure also allows the designer to simultaneously monitor several nodes of a circuit. In other words, the designer may use one channel of an oscilloscope to monitor multiple output waveforms, as shown in Figure 5.6. Moreover, the use of the ABIST structure for such implementation, particularly for AC analysis, may have a minor problem in phase shifting due to the clock frequency and applied frequency. More specifically, consider an n-stage ABIST. Let f_s be the applied frequency of the circuit under test and f_c be the clock frequency of the loading switch. The delay time T_{d_k} and phase shifting Q_{d_k} of kth stage input can be predicted as follows

$$T_{d_k} = \frac{n-k}{n+1}\frac{1}{f_c} \tag{5.1}$$

$$Q_{d_k} = 2\pi f_s T_{d_k} = 2\pi \frac{n-k}{n+1}\frac{f_s}{f_c} \tag{5.2}$$

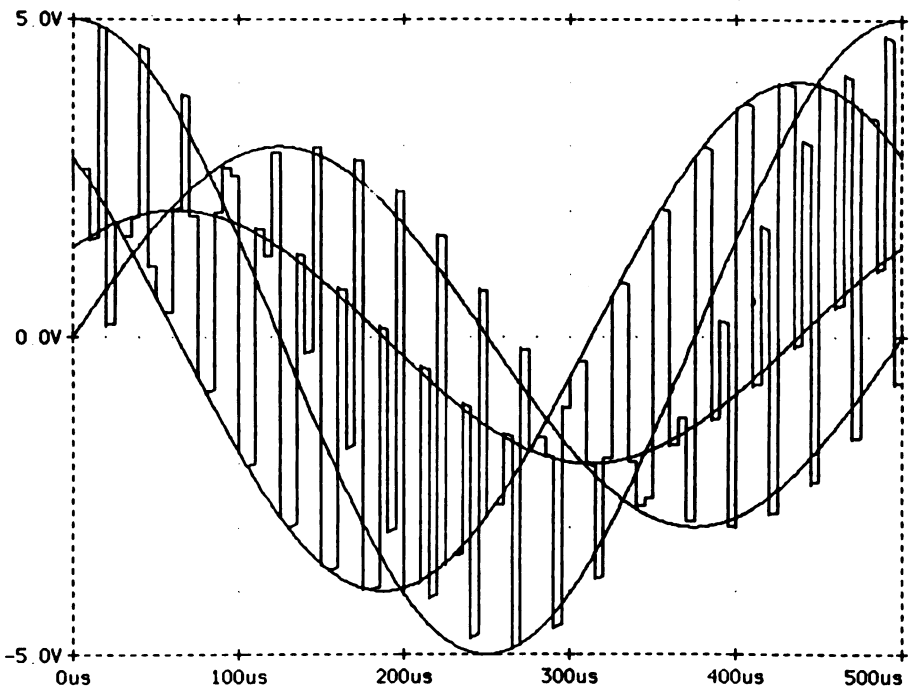
Consider the output waveform of Figure 5.7(b), where f_s=1kHz and f_c= 40kHz, that is t_c = 25μs. For a four stage ABIST structure, n=4, the delay time T_{d_k}=(4–k)5μs. For example, the delay time of the second stage is T_{d_2} = 10 μs. In order to illustrate the delay time of each stage, part of Figure 5.7(b) is shown in Figure 5.7 (c). In Figure 5.7 (c), the point b is at the end of loading time of the ABIST structure, and the point b' is the time when the test data held in the second stage is observed at *SO* pin. Therefore, the time difference $T_{d_2} = t_b - t_{b'}$ = 10μs, which matchs the the results calculated by Equation (5.1).

Similarly, T_{d_1}=15μs, T_{d_3}=5μs, and T_{d_4}=0μs. In other words, the stage that is closest to the output pin *SO* has least phase shifting. In addition, Equation (5.2) shows that the phase shifting is also dependent on the applied frequency f_s and clock frequency f_c. Decreasing f_s or increasing f_c will decrease the phase shifting. Of course, no phase shifting occurs in the DC case, due to f_s=0.

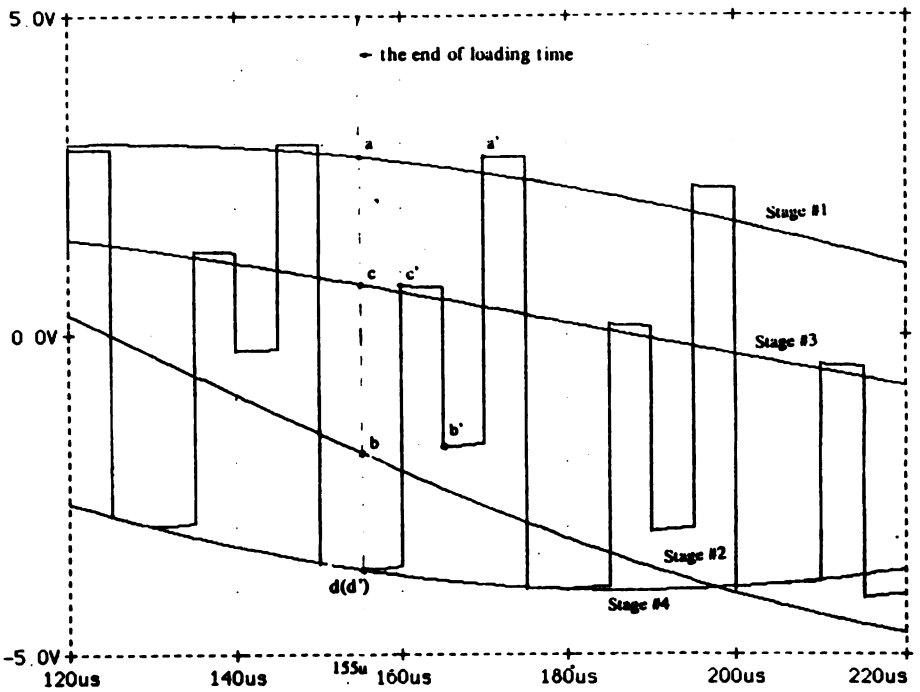


(a)

Figure 5.7 PSPICE Simulation Results: (a) DC;
(b) AC; and (c) a portion of (b).
(continued on next page)

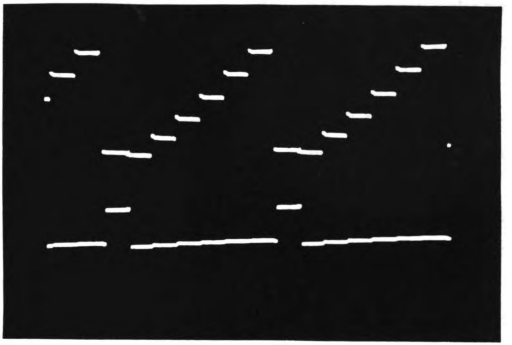


(b)

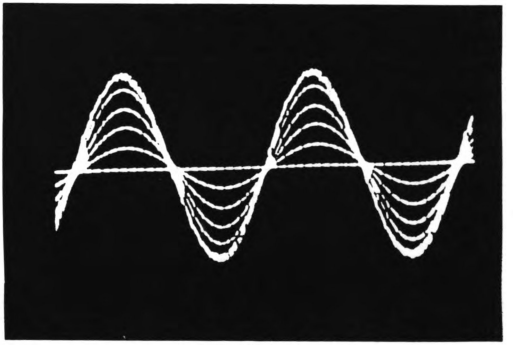


(c)

Figure 5.7 (Continued)



(a)



(b)

Figure 5.8 Observed Outputs from an Oscilloscope: (a) DC; and (b) AC.

5.4 VLSI IMPLEMENTATION

In this section, the design and layout of the two basic components, analog switches and voltage followers, for VLSI implementations of the ABIST structure are discussed.

5.4.1 Analog Switches

Figure 5.9 shows a CMOS analog switch. It is controlled by a pair of complementary clock signals ϕ and $\bar{\phi}$. When ϕ is low, both transistors are off, creating an effective open circuit. When ϕ is high, both transistors are on, giving a low impedance state. The ON resistance of the CMOS switch can be lower than $1k\Omega$ while the OFF leakage current is in the ten picoampere range. The bulk potentials of the p-channel V_{BP} and the n-channel V_{BN} are taken to the highest and lowest potentials, respectively.

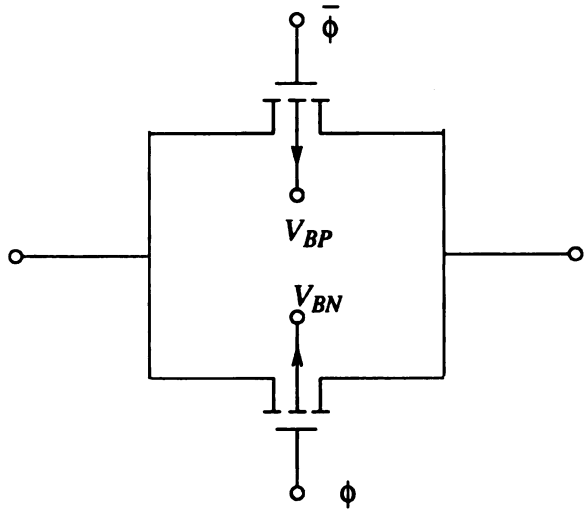


Figure 5.9 CMOS Analog Switch.

The CMOS switch has two advantages over the single-channel MOS switch . The first advantage is that the dynamic analog-signal range in the ON state is greatly increased. The second one is that since the n- and p-channel devices are in parallel and

require opposing clock signals, the feedthrough due to the clock will be diminished in some cases through cancellation.

5.4.2 Voltage Followers (VF)

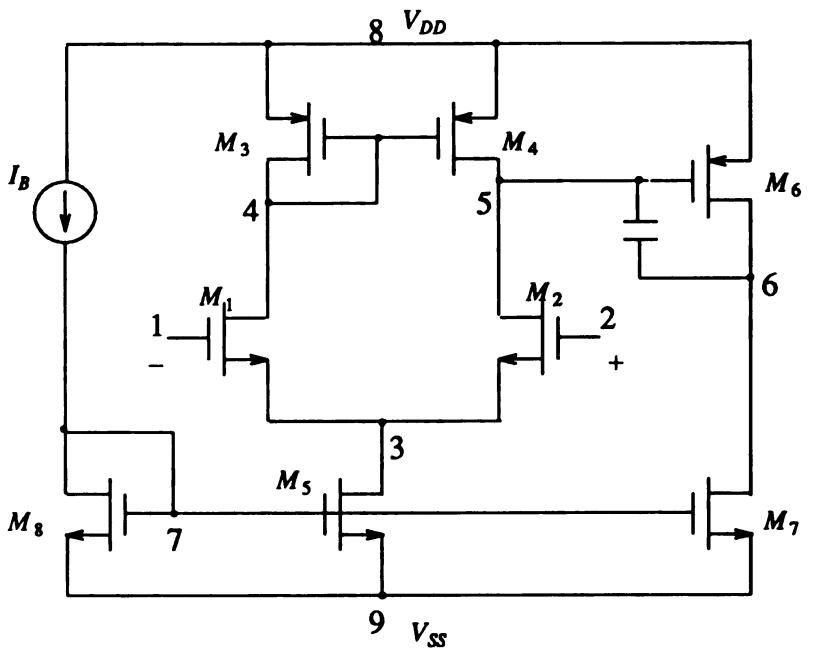


Figure 5.10 CMOS Op-amp with N Channel Input.

Figure 5.10 shows a voltage follower that is implemented by a CMOS unity-gain op-amp. The aspect ratios (W/L) of transistors and the capacitance are determined by the design specification and device parameters. They can be calculated according to the design procedure discussed in [1]. For example, consider the device parameters, as listed in Table 5.1, and the following design specification.

$A_v > 4000$ C_L =2pF V_{DD}=6V V_{SS}=-6V

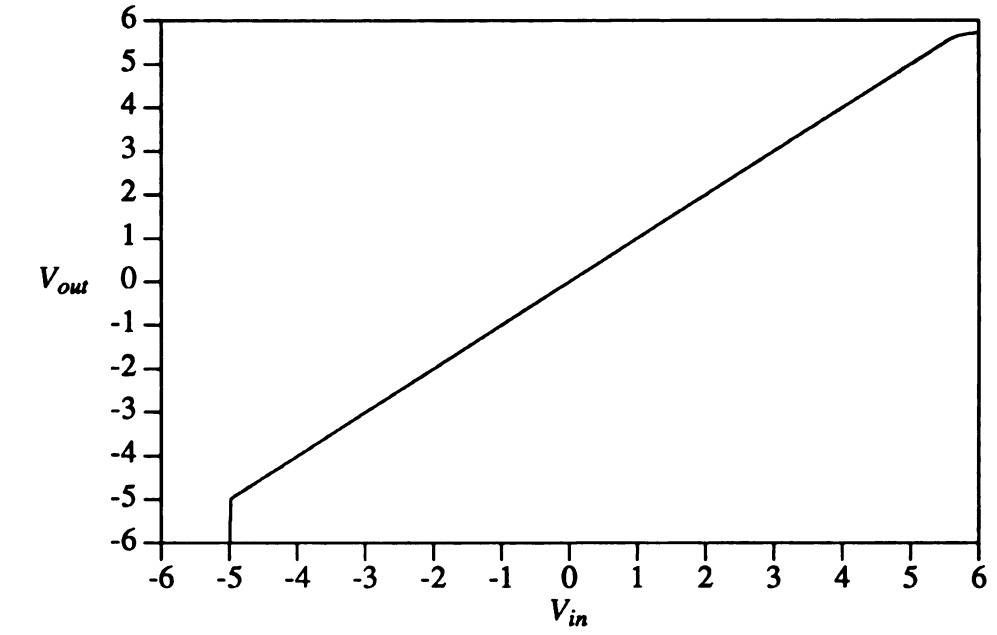
GB = 1MHz SR = 10v/μs V_{out} range = ±4V

CMR = ± 3V, P_{diss} < 10 mW, Channel length = 4.5 μm

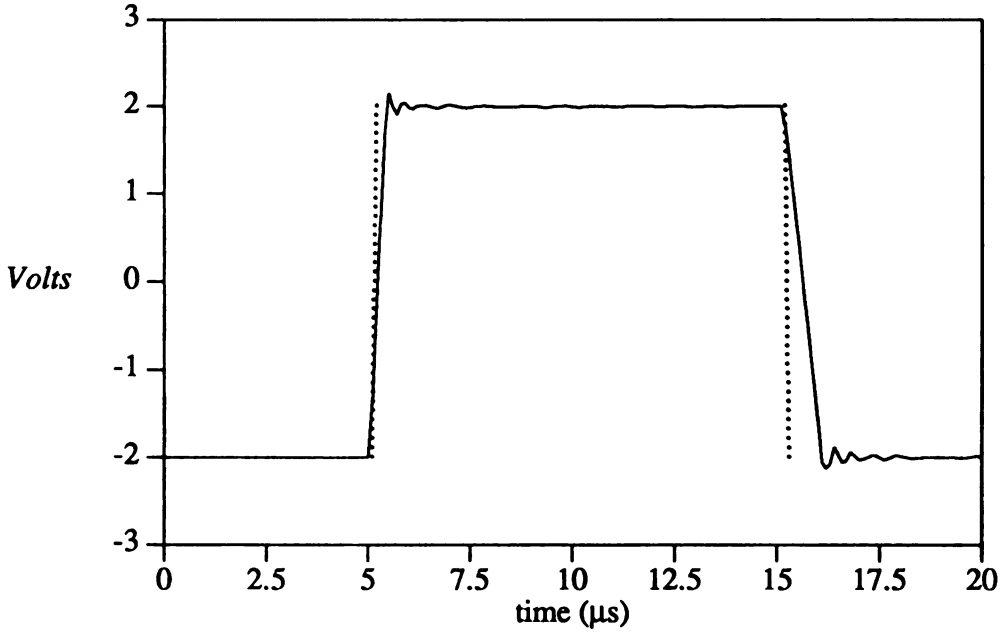
Table 5.1 Device Parameters

Parameter Symbol	Parameter Description	Typical Parameter Value		
		NMOS	PMOS	Unit
V_{TO}	Threshold Voltage ($V_{bs} = 0$)	0.83	-0.89	*volts*
K_{sat}'	Transconductance parameter (in saturation)	32.9	15.3	$\mu A/volt^2$
γ	Bulk Threshold parameter	1.36	0.88	$(volts)^{1/2}$
$2\lvert\phi_F\rvert$	Surface potential at strong inversion	0.6	0.6	*volts*
LD	Lateral diffusion	0.28	0.28	μm

The aspect rations (W/L) of the transistors and the compensation capacitance C_c are calculated in Appendix 3, and listed in Table 5.2. The op-amp design has been simulated by SPICE, where the SPICE input file is listed in Appendix 4. The simulation results for input CMR, frequency response and slew rate are plotted in Figure 5.11, and also listed in Table 5.3.

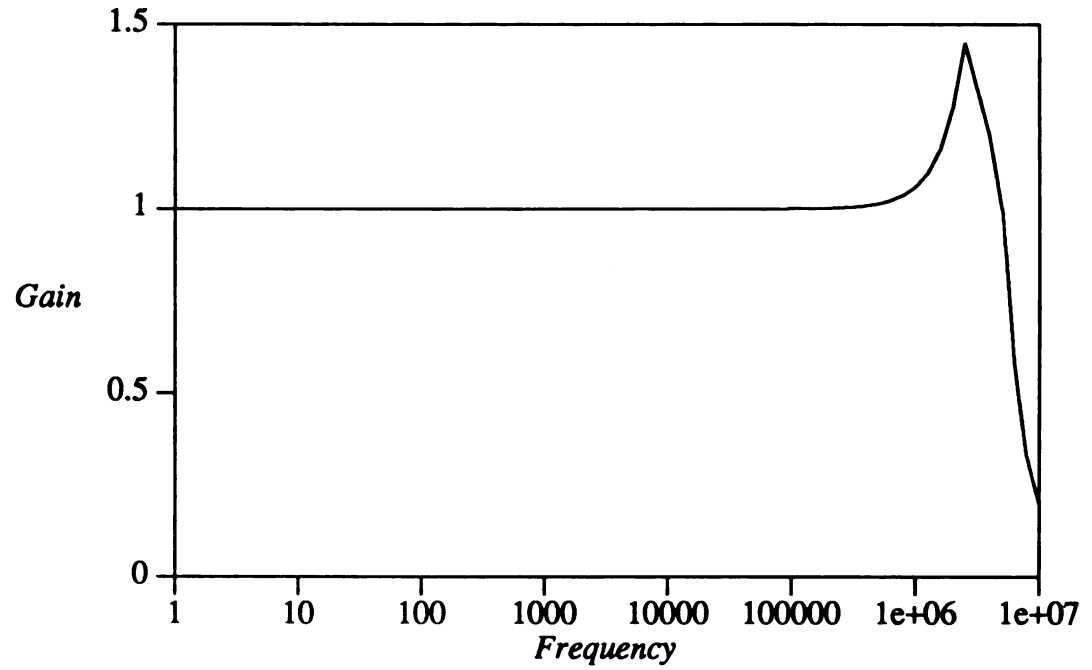


(a)

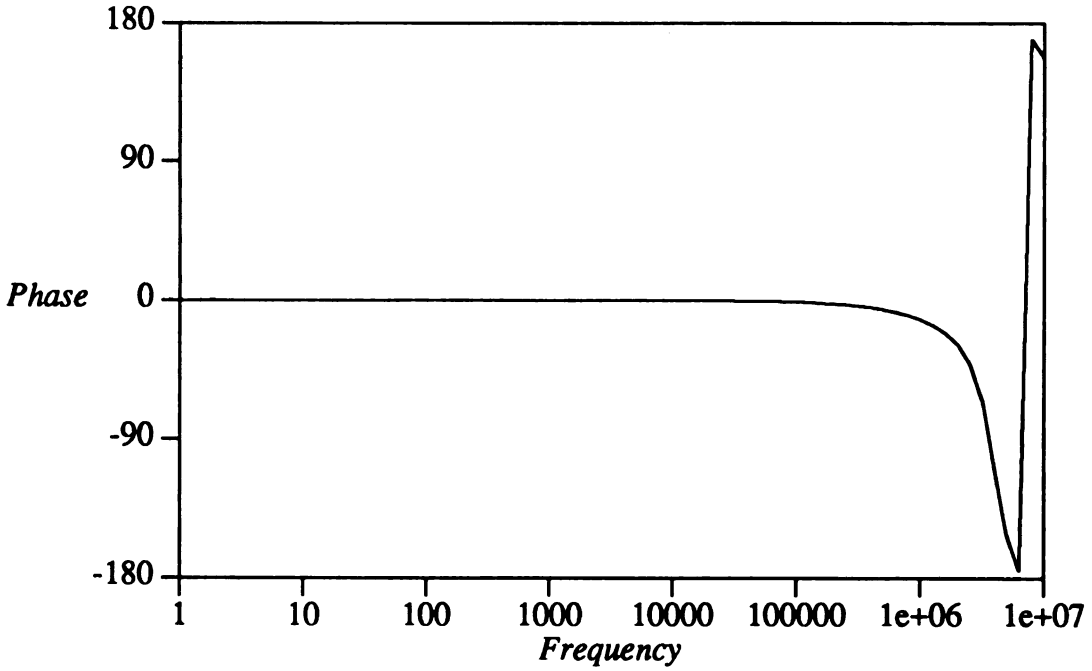


(b)

Figure 5.11 Simulation Results: (a) Input CMR Simulation; (b) Transient Response; (c) Magnitude Response; and (d) Phase Response. (Continued on next page)



(c)



(d)

Figure 5.11 (Continued).

Table 5.2 Device Area

Aspect ratio (W/L)	μm/μm
M1,M2	4.5/4.5
M3,M4	4.5/4.5
M5	4.5/4.5
M6	20/4.5
M7	10/4.5
M8	4.5/4.5
Compensation capacitance	0.44pF

Table 5.3 Simulation Result of an Op-amp Design

Specification	Design	Simulation
Unity gain	1	1
Unity gain bandwidth (MHz)	1	1
Input CMR (V)	±3	+5.4,–5
Slew Rate (V/μsec)	10	10
P_{diss} (mW)	<10	0.18
V_{DD},V_{SS} (V)		±6
Output resistance (Ω)	-	208
Input resistance (Ω)	-	10^8
Load capacitance (pF)	2	2

Table 5.3 shows that the op-amp design has a very low output resistance (208 Ω) and a very high input resistance (10^8Ω), it is well suitable for the use of the both voltage follower and test point buffer.

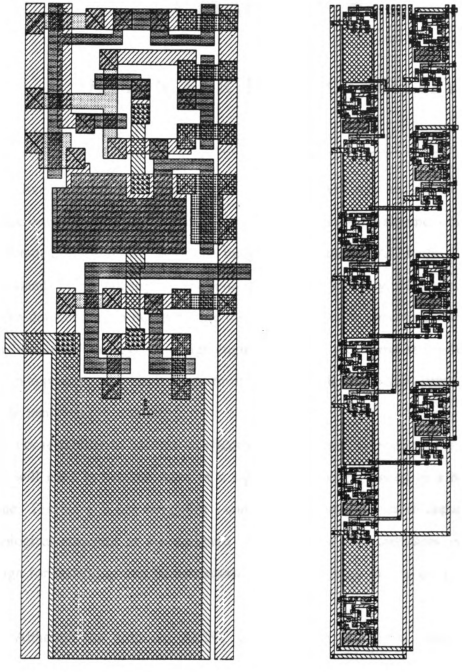
It should be noted that the input CMR is an important parameter for the implementation of ABIST structure. The input CMR (*Common Mode Range*) means that the unity gain voltage is held within the range specified. In other words, if the input voltage is out

of this range, then the output voltage will not be equal the input voltage. Since the system power supplies are ±5V, the voltages measured at various test points are also within this range. As a result, the linearity of the input CMR insures the unity-gain of the op-amp. However, the only drawback of this design is that the power supplies for the op-amp are ±6V. The use of the extra power supplies is inevitable. If we use the same power supplies for the op-amp, i.e., ±5V, the input CMR is ranged from –4V to +4.2V, which would loss the linearity at both ends. As an result, the measured data are inaccurate.

5.4.3 Physical Layout

Figure 5.12 shows a 4-stage ABIST structure, which has been layouted using 3 micron CMOS technology. This physical layout follows the design rules supported in Magic on SUN3 workstations.

The design began with the design of basic cells such as the CMOS op-amp and CMOS switch. Then, an one-stage ABIST module is formed in terms of these basic cells with necessary modification and interconnection, which is shown in Figure 5.12 (a). Finally, the 4-stage ABIST are constructed by duplicating the one-stage ABIST module and adding the routing to I/O connection. The complete layout for 4-stage ABIST structure is shown in Figure 5.12 (b). The total chip area is 143 λ by 673 λ. The scale factor is 1.5 μm / λ, so the size is 216537 μm^2. The ABIST structure has not yet been fabricated at this moment. The layout is given to show the approximate dimension of the op-amp design.



(a) (b)

Figure 5.12 The ABIST Structure (a) one-stage; and (b) 4-stage.

CHAPTER 6

DESIGN FOR DIAGNOSABILITY

Given the fact that future electronic systems will rely heavily on CAD tools to reduce design costs, increase design accuracy, and reduce development time, it is clearly important that testability factors be integrated into these CAD tools. Testability as it now stands in the industry is a "bottom-up" process. Virtually, all the known techniques need a detailed circuit design before the testability can be measured. Design, however, is a "top down" process. A design engineer is given requirements for design, creates a design approach, analyzes it for performance, finally is involved in structural design that can be analyzed for testability.

For testability to become an effective part of the CAD process, the testability process will need to become more of a top-down approach instead of a bottom-up technique. Therefore, if one can define a condition for testability which depends only on the topological structure of the designed circuit, not on the component values, then the design for testability can be established before analyzing the circuit performance [15, 45].

Although the proposed ABIST structure may provide as many test points as required for a circuit under test, the number of test points should not be indefinitely increased because of the placement and routing problems. It is necessary to select an appropriate minimum set of test points that is sufficient to diagnose a circuit. In this chapter, based on the above testability design principle, the issues of diagnosability and

test set generation are addressed.

The former issue concerns the diagnosability measurement of a circuit with a given set of test points. On the other hand, the latter issue deals with selecting an appropriate set of test points to meet the required diagnosability. These two issues are clearly important to the design engineers, especially for highly complex circuit designs.

6.1 DIAGNOSABILITY MEASUREMENT

According to the decomposition approach discussed in Chapter 4, a large network is decomposed into smaller subnetworks and its decomposition nodes are taken as the initial set of test points. The voltages measured at these test points are applied to check whether or not the network is t-diagnosable. A network is said to be *t-diagnosable* if, under the assumption that the network contains at most t faulty subnetworks, all the faulty subnetworks can be located.

Let $NS=(NS_{ij})_{n\times m}$ be a diagnostic matrix, where n and m are the numbers of decomposition nodes and subnetworks, and each entry NS_{ij} is defined as follows,

$$NS_{ij} = \begin{cases} 1 & \text{if the jth subnetwork is connected the ith decomposition nodes} \\ 0 & \text{otherwise} \end{cases}$$

Theorem 6.1 [31]

A Network is t-diagnosable if and only if the columns of matrix

$$D=[D_1, D_2, \cdots D_t]$$

are all distinct, where $D_1 = NS$ and D_i is a $n\times \binom{m}{i}$-matrix, whose column is constructed by logical ORing of any i columns of NS.

Example 6.1

Consider a network shown in Figure 6.1. The network is decomposed into 5 sub-networks: S_1, S_2, S_3, S_4 and S_5.

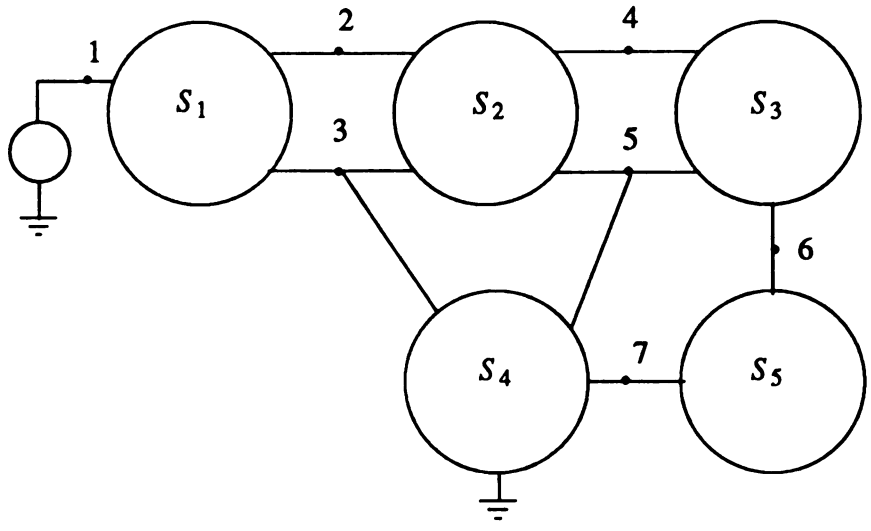


Figure 6.1 The Network in Example 6.1.

The diagnostic matrix *NS* is formed as follows

$$NS = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{bmatrix} S_1 & S_2 & S_3 & S_4 & S_5 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

This network is 1–*diagnosable*, because the columns of matrix *NS* are all distinct.

$$D=[D_1,D_2]=\begin{array}{c}\\ 1\\ 2\\ 3\\ 4\\ 5\\ 6\\ 7\end{array}\begin{bmatrix} S_1 & S_2 & S_3 & S_4 & S_5 & S_{12} & S_{13} & S_{14} & S_{15} & S_{23} & S_{24} & S_{25} & S_{34} & S_{35} & S_{45} \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{6.1}$$

where S_{ij} means S_iS_j, or logical ORing of S_i and S_j. Since the columns of matrix D are all distinct, by Theorem 6.1, the network is also 2-diagnosable.

For a given set of test points, Theorem 6.1 provides a simple way to check the diagnosability of the circuit. In practice, however, the dimension of the matrix D is grown with the complexity of the circuit. In order to reduce the memory space and computational complexity, Theorem 6.1 is modified as follows.

Theorem 6.2

A network is t-diagnosable, $t\geq 2$, if and only if it is $(t-1)$-diagnosable and the columns of matrix $[D_{t-1},D_t]$ are all distinct.

The proof of Theorem 6.2 is shown in Appendix 5. Theorem 6.2 checks the diagnosability recursively.

6.2 TEST POINTS SELECTION

Suppose that a t-diagnosable circuit network is designed, and the decomposition nodes are taken as the initial set of test points. Theorem 6.2 is applied to measure the diagnosability of this circuit, say, k-diagnosability. If $k\geq t$, then the set of test points is sufficient to diagnose the network. On the other hand, if $k<t$, more test points are needed to increase the desired diagnosability.

According to Theorem 6.2, a network is $(k-1)$-diagnosable, but not k-diagnosable, means that some columns of $[D_{k-1},D_k]$ are identical. Mathematically, one way to distinguish two identical columns of the matrix $[D_{k-1},D_k]$ is to add an additional row that contains different bit patterns in these two columns. Similarly, if there are r identical columns in that matrix, the use of q additional rows can make the identical columns all distinct, where $\lceil \log_2 r \rceil \le q \le r$.

In order to find the minimum set of test points, two steps of the test point selection process are implemented: *selection* and *compaction*. The former step is to identify those groups containing the identical columns and to add some additional rows. The latter step is employed to compact the row patterns selected in the first step.

Example 6.1 (continued)

Using the matrix D_1, D_3 is constructed as follows

$$D_3 = \begin{array}{c|cccccccccc} & S_{123} & S_{124} & S_{125} & S_{134} & S_{135} & S_{145} & S_{234} & S_{235} & S_{245} & S_{345} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 4 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 5 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 6 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 7 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}$$

We can easily find that some columns of matrix $[D_2,D_3]$ are identical. The groups of identical columns are listed in Table 6.1. By Theorem 6.2, this implies that the network is 2-diagnosable, but not 3-diagnosable.

In order to design a 3-diagnosable circuit, some additional test points are needed. Specifically, to distinguish the identical columns in each group listed in Table 6.1, we may simply add $\lceil \log_2 r \rceil$ rows to each group, where r is the number of identical columns in that group. For instance, we may add 2 extra rows for group 3. In other words, it requires at least 6 additional rows in this example for 3-diagnosability. However, it is possible to merge some rows to compact the test point set.

Table 6.1 Groups with Identical Columns in Example 6.1

Groups	Identical columns
1	S_{13}, S_{123}
2	S_{34}, S_{345}
3	$S_{125}, S_{134}, S_{135}$
4	$S_{25}, S_{234}, S_{235}, S_{245}$

Table 6.2 Test Point Selection and Compaction

columns	1	2	3	4	5	group
S_{13}	1	0	1	0	0	1
S_{123}	1	1	1	0	0	
S_{34}	0	0	1	1	0	2
S_{345}	0	0	1	1	1	
S_{125}	1	1	0	0	1	3
S_{134}	1	0	1	1	0	
S_{135}	1	0	1	0	1	
S_{25}	0	1	0	0	1	4
S_{234}	0	1	1	1	0	
S_{235}	0	1	1	0	1	
S_{245}	0	1	0	1	1	

A compaction step is followed. We first list the elements contained in each group as shown in Table 6.2. In group 1, for instance, S_{123} means the logical ORing of S_1, S_2, and S_3, we mark "1" at bits 1, 2, and 3, and "0" to others. Since the bit patterns of S_{13} and S_{123} are different in bit 2, hence, bit 2 can be used to distinguish these two identical columns in Group 1. Similarly, S_{125}, S_{134}, and S_{135}, in Group 3, can be distinguished

by bits 2 and 5, or bits 3 and 4, or bits 3 and 5. Therefore, the compaction problem is turned into the following optimization problem.

$$G = \text{Min} \bigcup_{i=1}^{g} \left\{ \min G_i \right\} \tag{6.2}$$

where G_i, i=1,2, $\cdots$,g, is a collection of all possible bit patterns that can distinguish all identical columns in Group i, and min G_i is a subset of G_i that contains all minimized bit patterns. (A *minimized bit pattern* is the pattern that is not contained in any other patterns in min G_i.) For example,

$$\min G_1 = \{ 2 \}$$

$$\min G_2 = \{ 5 \}$$

$$\min G_3 = \{ \{ 2, 5 \}, \{ 3, 4 \}, \{ 3, 5 \} \}$$

$$\min G_4 = \{ \{ 3, 4 \} \}$$

Therefore, G = { 2, 3, 4, 5 }. In other words, four extra test points are needed to design a 3-diagnosable circuit. The new matrix $[D_2,D_3]$ is formed as follows,

$[D_2,D_3]$

	S_{12}	S_{13}	S_{14}	S_{15}	S_{23}	S_{24}	S_{25}	S_{34}	S_{35}	S_{45}	S_{123}	S_{124}	S_{125}	S_{134}	S_{135}	S_{145}	S_{234}	S_{235}	S_{245}	S_{345}
1	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
2	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0
3	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
4	1	1	0	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1
=5	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	0	1	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
7	0	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
a	1	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	0
b	0	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	1	0	1
c	0	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	1
d	0	0	0	1	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	1

where the rows a,b,c, and d are the additional test points and they are respectively representing the elments of G. These rows are constructed as follows. For the row a, all columns in $[D_2,D_3]$ with "2" in subscripts of S are marked to "1" and others are "0". Similarly, all columns with subscripts "3", "4", and "5" are marked to "1" to the rows

b,*c*, and *d*, respectively.

The above example shows that the circuit is 3-diagnosable if the above 4 extra test points are added. Interestingly, with the following matrix D_4,

$$D_4 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ a \\ b \\ c \\ d \end{array} \begin{bmatrix} S_{1234} & S_{1235} & S_{1245} & S_{1345} & S_{2345} \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

we also find that the network is 4-diagnosable without adding any other additional test points.

The selection and compaction processes discussed above is summarized as follows.

Procedure Select-Compact()

Step 1. (Selection)

1.1 Identify the groups of identical columns in $[D_{k-1},D_k]$, and construct the table as Table 6.2.

1.2 Find min G_i for each group.

Step 2. (Compaction)

2.1 Find G of (6.2). Say, $G = \{ g_1, \cdots, g_p \}$

2.2 For i=1 to p

Add a row R_i to matrix $[D_{k-1},D_k]$

mark 1 to those column with subscript g_i

Step 3. The circuit is *k*-diagnosable.

The entire test points selection procedure is summarized as follows.

Algorithm IV

Step 1. Construct the diagnostic matrix NS ($= D_1$), and set $k=1$.

If columns of D_1 are all distinct, GOTO Step 2;

otherwise GOTO Step 3.

Step 2. (Diagnosability Measurement)

If ($k \geq t$) GOTO Step 4, otherwise $k=k+1$; and construct D_k.

If some columns of $[D_{k-1}, D_k]$ are identical, GOTO Step 3;

otherwise, repeated this step.

Step 3. (Test Point Selection and Compaction)

Call procedure *Select-Compact*() to make the circuit k-diagnosable.

GOTO Step 2.

Step 4. The circuit is t-diagnosable.

Example 6.2

Figure 6.2 shows a ten band octave equalizer, where nodes 1, 3, 4, and 6 are chosen as decomposition nodes. The circuit is decomposed into 12 subnetworks, as shown in Figure 6.3.

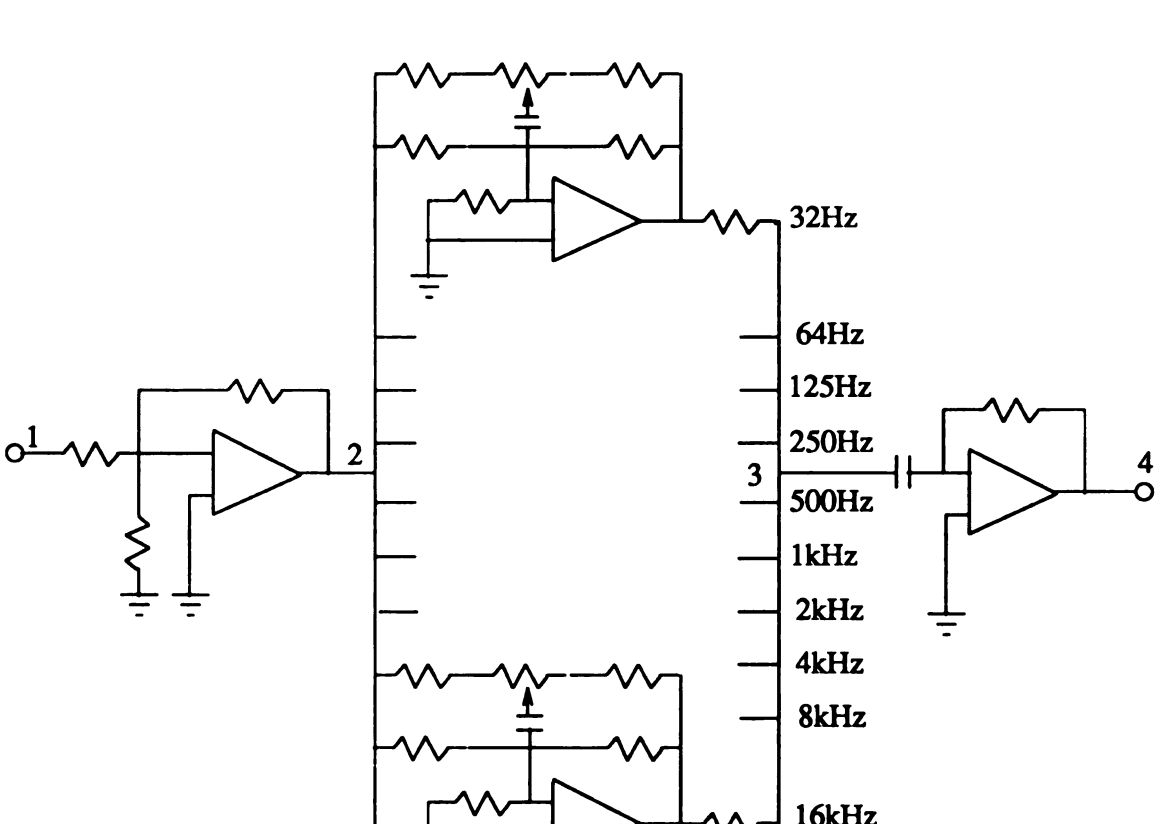


Figure 6.2 A Ten Band Octave Equalizer.

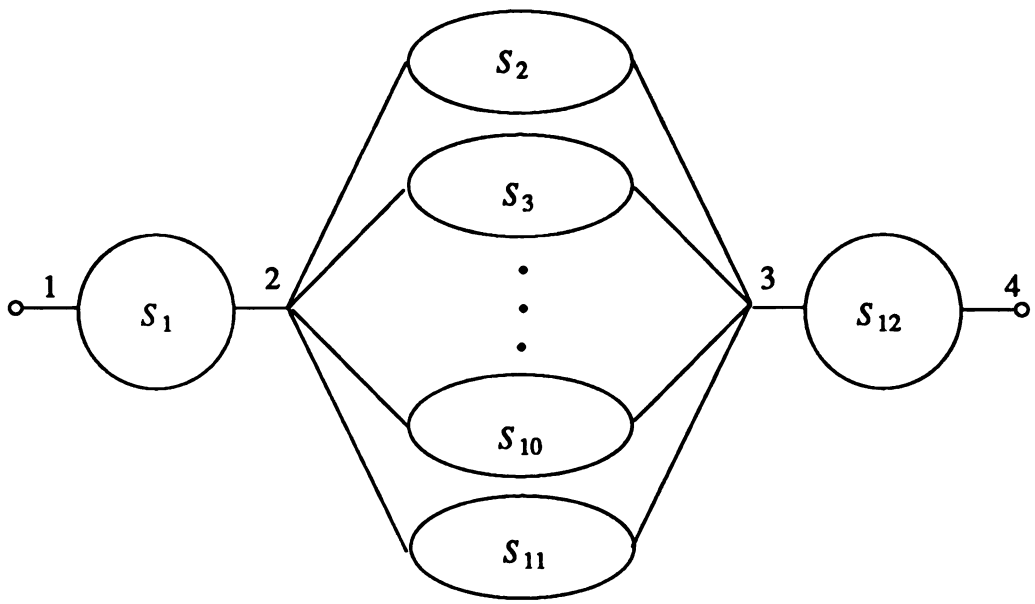


Figure 6.3 A Decomposition of the Octave Equalizer.

We first construct the *NS* matrix for this circuit.

$$NS = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{bmatrix} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} & S_{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Since columns S_2 through S_{11} are identical, the circuit is not 1-diagnosable. Following Algorithm IV, 10 extra test points are added as follows.

$$NS = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \\ j \end{array} \begin{bmatrix} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} & S_{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Since all columns are distinct, the network is 1-diagnosable. With such set of test points, the network is also 12-diagnosable.

CHAPTER 7

CONCLUSIONS

This chapter summarizes the major contribution of this dissertation research and outlines the directions for future research.

7.1 SUMMARY OF MAJOR CONTRIBUTIONS

For more than two decades, the subjects of the automatic testing and fault diagnosis of electronic circuits have been of interest to researchers in the areas of circuits and systems. Recently, with the rapidly increasing complexity and size of modern electronic systems, these subjects become more and more important and critical.

A fault location algorithm and an AATPG (Analog Automatic Test Program Generator) have been developed to precisely locate faults in analog circuits.

For a reliable design, potential faulty components in a system should be identified as soon as possible and be replaced before the system fails. A fault prediction problem has been initiated and a fault prediction algorithm has been presented, which can predict if any of the network components is about to fail, so that the potential faulty components can be predicted and replaced before the system actually fails.

In order to diagnose large circuit networks with reasonably high speed, a decomposition approach fault prediction algorithm is proposed. The algorithm can be used hierarchically to decompose a network into any desired level. In order to precisely locate the faults at a specific level, it is necessary to provide sufficiently many accessible

nodes for voltage measurement. Unfortunately, modern electronic systems are often multi-layered and/or coated, thereby limiting the accessibility of test points. It is difficult to provide proportionately more I/O connections as the number of components in a unit increases. This becomes the bottleneck of analog fault diagnosis problem.

In this dissertation, an Analog Built-In Self-Test (ABIST) structure has been proposed which provides more test points while still keeping low pin overhead. It is the first Built-In Self-Test structure for analog fault diagnosis. Included in this study are topological structure and operation of the ABIST structure, simulation results and a hardware implementation indicating the principle works well, and the design and simulation of the basic ABIST components. A detailed layout of the structure has been done using 3 micron CMOS technology. The ABIST structure can be used to acquire test data at various test points simultaneously, a very important requirement for diagnosis. In addition to the fault diagnosis use, the proposed ABIST structure also allows system designers to use one channel of an oscilloscope to simultaneously monitor multiple output waveforms of a circuit or system.

Although the proposed ABIST structure may provide as many test points as required for a circuit under test, the number of test points should not be indefinitely increased because of the placement and routing problems. It is necessary to select an appropriate minimum set of test points that is sufficient to diagnose a circuit. Therefore, an efficient algorithm to select test points for the design of a t-diagnosable network has been proposed.

In summary, this dissertation focuses on fault prediction and diagnosis in large circuit networks. It is novel in the sense that it is the first analog BIST structure ever proposed for analog fault diagnosis. This structure will be very helpful in developing efficient and practical fault diagnosis algorithms for large circuit networks.

7.2 DIRECTIONS FOR FUTURE RESEARCH

Some practical issues in analog fault diagnosis are addressed here for further study.

7.2.1 Automatic Testing and Diagnosis

In this dissertation, an ABIST structure and an AATPG software package have been presented. However, the development of an Automatic Test Equipment (ATE) [24], required for automatic testing and diagnosis, has not yet been discussed. Basically, the functions that can be performed by the ATE include (1) supply test stimuli; (2) obtain output response; and (3) diagnose faults. A block diagram of an ATE system is shown in Figure 7.1. A microprocessor acts as the center of the ATE, which controls the stimulus unit to stimuli the input signal to UUT and sends a clock signal to the clock circuit in order to form a set of proper clock signals to control analog switches of the ABIST structure. The analog test data acquired from pin SO of the ABIST structure are first converted to digital signals by the A/D converter, which are then sent to the microprocesser and transferred as the input data of AATPG software package [16, 17, 18]. AATPG is used to diagnose faults.

7.2.2 Time Domain Analysis

The proposed AATPG software package can handle both DC and AC analysis for linear circuits very well. For nonlinear circuits, however, the SPICE is employed to handle the transient analysis in time domain. In practice, it is difficult to directly send test data to the SPICE program at every time step. This poses an interesting research topic for either developing a transient analysis program for time domain, or modifying SPICE by adding an interface program.

7.2.3 Fault Diagnosis for Analog/Digital Hybrid Circuit

Typically, a practical circuit may consist of both analog and digital components. Several fault diagnosis algorithms for digital circuit, or analog circuits, have been

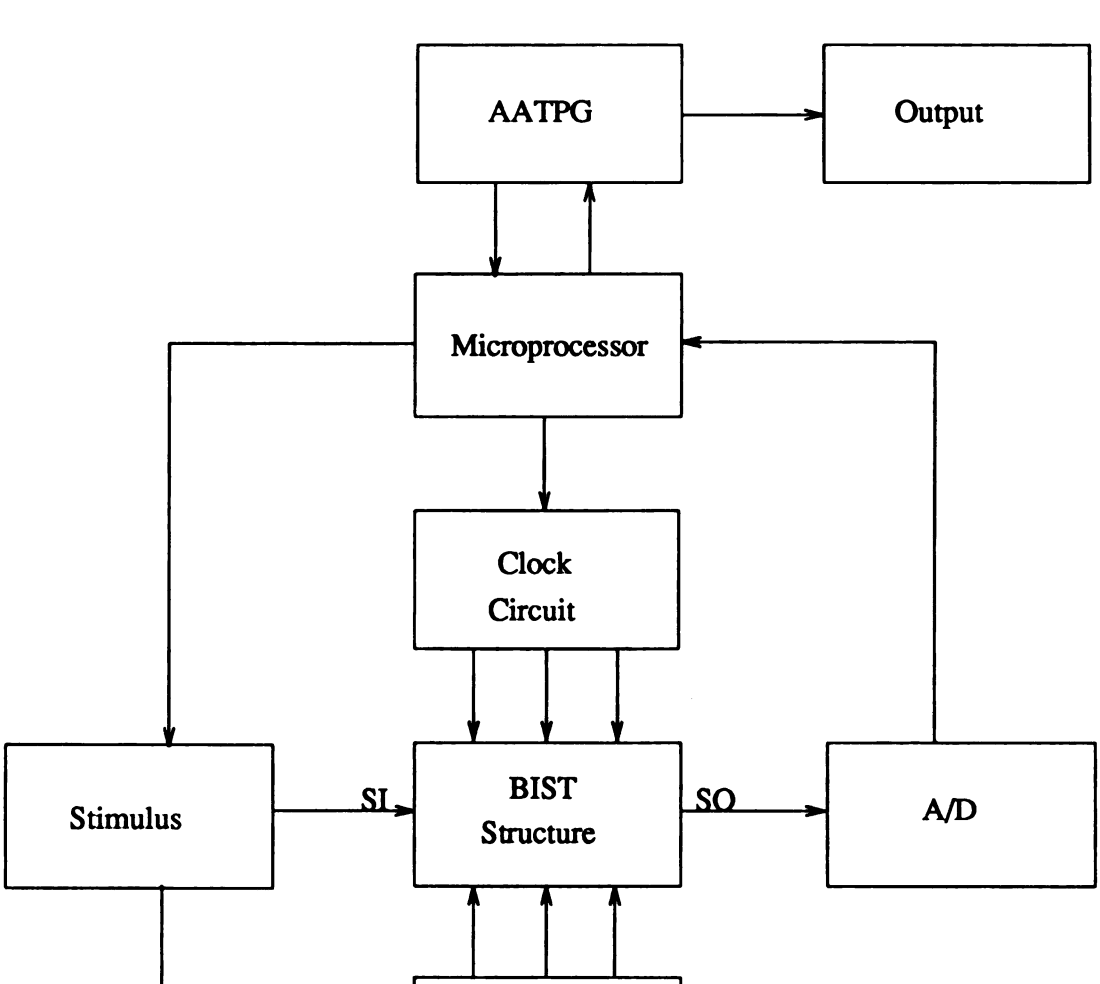


Figure 7.1 A Block Diagram of ATE.

developed, but, there are no general methods available for fault diagnosis of analog/digital hybrid circuit [35]. Since the proposed ABIST structure provides the possibility to acquire test data for the analog/digital hybrid circuit, the structure will be very helpful to develop an efficient fault diagnosis algorithm for the analog/digital hybrid circuit.

7.2.4 BIST Implementation

Unlike the digital BIST structure used to input test patterns and acquire output responses, the proposed ABIST is only employed to acquire test data. In some analog diagnosis algorithms, applying external excitation to some test points is required. In that case, the proposed ABIST structure can be modified to apply the required excitation.

According to the switching operation of the proposed ABIST structure shown in Figure 6.4, the datum holding at first stage of the buffers will be passed out last. As a result, the capacitance required in that stage must be sufficient enough to hold the datum before it is passed out. In other words, the greater the number of stages, the greater the capacitance that must be designed. However, the greater capacitance requires more chip area. For this reason, it is necessary to provide a precise measurement as a criterion to determine the number of stages.

In practice, it is possible to lower the capacitance by reducing the number of stages. This can be accomplished by using more scan out pins. The structure is suggested in Figure 7.2, where each lane has its own scan out pin and connects to an A/D converter. There is a trade-off between the chip area and pin overhead.

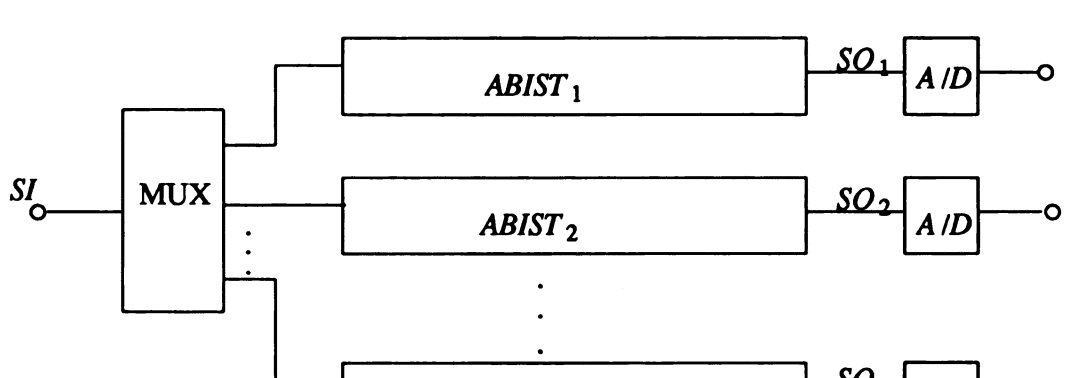


Figure 7.2 ABIST array.

APPENDICES

APPENDIX 1

Equivalent Fault Set

Property 2.1

If the actual fault set contains a subset consisting of either

a) l components in loop L which consists of $l+1$ components, or

b) l components in cutset C which consists of $l+1$ components incident to an inaccessible node,

then an f-component-set formed by a combination of any l components of loop L or cutset C together with other $f-l$ components of the actual fault set, is an equivalent fault set.

Proof: **a)** Assume that components of the actual fault set are numbered as $(1,2,\cdots,l,q,\cdots,f)$, where the faulty components, $1,2,\cdots$, and l are in loop L which consists of $l+1$ components, the other faulty components are not in this loop. For a actual fault set, we have

$$W_{f\,(act)} = \begin{bmatrix} \hat{v}_1^{(1)} & \cdots & \hat{v}_j^{(1)} & \cdots & \hat{v}_l^{(1)} & \hat{v}_q^{(1)} & \cdots & \hat{v}_f^{(1)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hat{v}_1^{(m)} & \cdots & \hat{v}_j^{(m)} & \cdots & \hat{v}_l^{(m)} & \hat{v}_q^{(m)} & \cdots & \hat{v}_f^{(m)} \end{bmatrix} \tag{1}$$

According to KVL, in a loop L,

$$\sum_{i=1}^{l+1} k_i \hat{v}_i = 0, \quad where \;\; k_i = \pm 1.$$

One of the component voltages is the linear combination of other l component voltages,

that is

$$\hat{v}_j = -\frac{1}{k_j}\sum_{i=1,i\neq j}^{l+1} k_i\hat{v}_i, \quad where \ \ j=1,2,\cdots,l+1. \tag{2}$$

Substituting equation (2) for $\hat{v}_j$ in equation (1)

$$W_{f\,(act)} = \begin{bmatrix} \hat{v}_1^{(1)} & \cdots & \hat{v}_{j-1}^{(1)} & -\frac{1}{k_j}\sum_{i=1,i\neq j}^{l+1} k_i\hat{v}_i^{(1)} & \hat{v}_{j+1}^{(1)} & \cdots & \hat{v}_l^{(1)} & \hat{v}_q^{(1)} & \cdots & \hat{v}_f^{(1)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hat{v}_1^{(m)} & \cdots & \hat{v}_{j-1}^{(m)} & -\frac{1}{k_j}\sum_{i=1,i\neq j}^{l+1} k_i\hat{v}_i^{(m)} & \hat{v}_{j+1}^{(m)} & \cdots & \hat{v}_l^{(m)} & \hat{v}_q^{(m)} & \cdots & \hat{v}_f^{(m)} \end{bmatrix}$$

Through l times column basic transformation, we get

$$\begin{bmatrix} \hat{v}_1^{(1)} & \cdots & \hat{v}_{j-1}^{(1)} & \hat{v}_{j+1}^{(1)} & \cdots & \hat{v}_l^{(1)} & \hat{v}_{l+1}^{(1)} & \hat{v}_q^{(1)} & \cdots & \hat{v}_f^{(1)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hat{v}_1^{(m)} & \cdots & \hat{v}_{j-1}^{(m)} & \hat{v}_{j+1}^{(m)} & \cdots & \hat{v}_l^{(m)} & \hat{v}_{l+1}^{(m)} & \hat{v}_q^{(m)} & \cdots & \hat{v}_f^{(m)} \end{bmatrix} = W_f^j$$

and

$$rank[W_f \mid \Delta P] = rank[W_{f\,(act)} \mid \Delta P]$$

Since ΔP is a linear combination of all columns of $W_{f\,(act)}$, and $W_{f\,(act)}$ is equivalent to W_f^j, thus ΔP is also linear combination of all columns of W_f^j. This implies that

$$rank[W_f^j \mid \Delta P] = rank[W_f^j] = f$$

b) According to the KCL, in a cutset C which incident to an inaccessible node,

$$\sum_{i=1}^{l+1} k_i\hat{i}_i = 0, \quad where \ \ k_i=\pm 1.$$

Assume $i=vg$, then

$$\sum_{i=1}^{l+1} k_i g_i\hat{v}_i = 0, \quad where \ \ k_i=\pm 1.$$

One of the component voltages is the linear combination of other l component voltages, that is

$$\hat{v}_j = -\frac{1}{k_j g_j} \sum_{i=1, i\neq j}^{l+1} k_i g_i \hat{v}_i, \quad where \ \ j=1,2,\cdots,l+1.$$

Through a similar deriving as that in (a), we get

$$rank[W_f^j \mid \Delta P] = rank[W_f^j] = f$$

Property 2.2

If $m-f+1$ row dependence of W_f is consistent with row dependence of $W_{f(act)}$, and $rank[W_f] = f$, the f-component-set corresponding to W_f is an equivalent fault set.

Proof: Assume that the first row of $W_{f(act)}$ is linearly dependent on the last $m-f$ rows, we have

$$W_{f(act)} = \begin{bmatrix} w_{1a}^{(1)} & w_{2a}^{(1)} & \cdots & w_{fa}^{(1)} \\ \cdot & \cdot & \cdot & \cdot \\ w_{1a}^{(f)} & w_{2a}^{(f)} & \cdot & w_{fa}^{(f)} \\ k_1 w_{1a}^{(1)} & k_1 w_{2a}^{(1)} & \cdot & k_1 w_{fa}^{(1)} \\ \cdot & \cdot & \cdot & \cdot \\ k_{m-f} w_{1a}^{(1)} & k_{m-f} w_{2a}^{(1)} & \cdots & k_{m-f} w_{fa}^{(1)} \end{bmatrix}$$

In the actual fault set, we know

$$\Delta p_1 = \sum_{j=1}^{f} w_{ja}^{(1)} \Delta x_j$$

and then

$$\Delta p_{f+i} = \sum_{j=1}^{f} w_{ja}^{(f+i)} \Delta x_j = \sum_{j=1}^{f} k_i w_{ja}^{(1)} \Delta x_j = k_i \sum_{j=1}^{f} w_{ja}^{(1)} \Delta x_j = k_i \Delta p_1, \quad \text{where } i=1,2,\cdots,m-f.$$

Therefore,

$$\Delta P = [\Delta p_1, \Delta p_2, \cdots, \Delta p_f, k_1 \Delta p_1, \cdots k_{m-f} \Delta p_1]^T$$

As the given condition, $m-f+1$ row dependence of W_f is consistent with row dependence of $W_{f(act)}$, then:

$$W_f\Delta P=\begin{bmatrix} w_1^{(1)} & w_2^{(1)} & \cdots & w_f^{(1)} & \Delta p_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ w_1^{(f)} & w_2^{(f)} & \cdot & w_f^{(f)} & \Delta p_f \\ k_1 w_1^{(1)} & k_1 w_2^{(1)} & \cdot & k_1 w_f^{(1)} & k_1 \Delta p_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ k_{m-f} w_1^{(1)} & k_{m-f} w_2^{(1)} & \cdots & k_{m-f} w_f^{(1)} & k_{m-f} \Delta p_1 \end{bmatrix}$$

Through $m-f$ times basic transformation, we have

$$W_f\Delta P=\begin{bmatrix} w_1^{(1)} & w_2^{(1)} & \cdots & w_f^{(1)} & \Delta p_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ w_1^{(f)} & w_2^{(f)} & \cdot & w_f^{(f)} & \Delta p_f \\ 0 & 0 & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

Thus, $rank W_f\Delta P = f$. Also, as the given condition $rank W_f = f$, we have

$$rank[W_f\Delta P] = rank W_f = f.$$

APPENDIX 2

PSPICE Code for ABIST Simulation

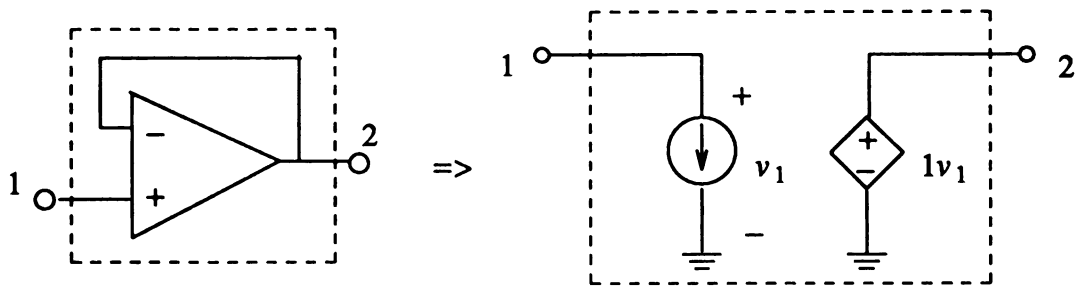


Figure a.1 Unity-gain Opamp Micromodel

```
A Four Stage ABIST Structure in DC
.probe
vc 5 0 0 pulse(0 5 0.01u 0.01u 0.01u 4.9u 25u)
rc 5 0 100meg
v0 4 0 2v
x0 1 2 opamp
x10 4 3 opamp
s0 1 3 5 0 on
c0 1 0 5p
*
v1 14 0 4v
x1 11 12 opamp
x11 14 13 opamp
s1 11 13 5 0 on
c1 11 0 5p
```

```
s11 2 11 16 0 on
vc1 16 0 0 pulse ( 0 5 20u 0.01u 0.01u 4.9u 25u)
rc1 16 0 100meg
*
v2 24 0 -1v
x2 21 22 opamp
x12 24 23 opamp
s2 21 23 5 0 on
c2 21 0 5p
s12 12 21 26 0 on
vc2 26 0 0 pulse ( 0 5 15u 0.01u 0.01u 9.9u 25u)
rc2 26 0 100meg
*
v3 34 0 3v
x3 31 32 opamp
x13 34 33 opamp
s3 31 33 5 0 on
c3 31 0 5p
s13 22 31 36 0 on
vc3 36 0 0 pulse ( 0 5 10u 0.01u 0.01u 14.9u 25u)
rc3 36 0 100meg
*
x4 41 42 opamp
s14 32 41 46 0 on
vc4 46 0 0 pulse ( 0 5 5u 0.01u 0.01u 19.9u 25u)
rc4 46 0 100meg
rl 42 0 10meg
*
.subckt opamp 1 2
i1 1 0 1p
ea 2 0 1 0 1
.ends
```

```
*
.model on vswitch (ron=1m roff=100meg von=4.9 voff=0.1)
.tran 1u 50u
.print tran v(42)
.end

A Four Stage ABIST Structure in AC
.probe
vc 5 0 0 pulse(0 5 0.01u 0.01u 0.01u 4.9u 25u)
rc 5 0 100meg
v0 4 0 sin (0 2 2000)
x0 1 2 opamp
x10 4 3 opamp
s0 1 3 5 0 on
c0 1 0 5p
*
v1 14 0 sin (0 5 2000 0 0 90)
x1 11 12 opamp
x11 14 13 opamp
s1 11 13 5 0 on
c1 11 0 5p
s11 2 11 16 0 on
vc1 16 0 0 pulse ( 0 5 20u 0.01u 0.01u 4.9u 25u)
rc1 16 0 100meg
*
v2 24 0 sin (0 2 2000 0 0 45)
x2 21 22 opamp
x12 24 23 opamp
s2 21 23 5 0 on
c2 21 0 5p
s12 12 21 26 0 on
vc2 26 0 0 pulse ( 0 5 15u 0.01u 0.01u 9.9u 25u)
```

```
rc2 26 0 100meg
*
v3 34 0 sin (0 4 2000 0 0 135)
x3 31 32 opamp
x13 34 33 opamp
s3 31 33 5 0 on
c3 31 0 5p
s13 22 31 36 0 on
vc3 36 0 0 pulse ( 0 5 10u 0.01u 0.01u 14.9u 25u)
rc3 36 0 100meg
*
x4 41 42 opamp
s14 32 41 46 0 on
vc4 46 0 0 pulse ( 0 5 5u 0.01u 0.01u 19.9u 25u)
rc4 46 0 100meg
rl 42 0 10meg
*
.subckt opamp 1 2
i1 1 0 1p
ea 2 0 1 0 1
.ends
*
.model on vswitch (ron=1m roff=100meg von=4.9 voff=0.1)
.tran 1u 500u
.option itl5=0
.end
```

APPENDIX 3

Design Procedure of a CMOS Opamp

Using the material and device parameter given in Table 5.1, according to the design procedure discussed in [1], design an amplifier similar to that shown in Figure 5.10 that meets the following specifications

$A_v > 4000$ C_L =2pF V_{DD}=6V V_{SS}=-6V

GB = 1MHz SR = 10v/μs V_{out} range = ±4V

CMR = ± 3V, $P_{diss} < 10$ mW, Channel length = 4.5 μm

1. Calculate the minimum value of the compensation capacitor C_c

$$C_c > 0.22C_L = (2.2/10)(2pF) = 0.44pF$$

2. Determine the minimum value for the "tail current" (I_5) from the slew-rate specification and C_c

$$I_5 = SR \times C_c = (0.44\times10^{-12})(10\times10^6) = 4.4\mu A$$

3. Design for $(W/L)_3$ from the maximum input voltage specification

$$(W/L)_3 = \frac{I_5}{K_3[V_{DD}-V_{in}(max)-|V_{TO3}|(max)+V_{T1}(min)]^2}$$

$$= \frac{4.4\times10^{-6}}{(15.3\times10^{-6})(6-3-0.89+0.83)^2} = 0.033$$

4. Design for $(W/L)_2$ to achieve the desired GB

$$gm2 = GB \times C_c = (2)(3.14)(1\times10^6)(0.44\times10^{-12}) = 2.765\mu S$$

$$(W/L)_2 = \frac{g_{m2}^2}{k_2 I_5} = \frac{(2.765\times10^{-6})^2}{(32.9\times10^{-6})(4.4\times10^{-6})} = 0.053$$

5. Design for $(W/L)_5$ from minimum input voltage. First calculate V_{DS5}(sat) then find $(W/L)_5$

$$V_{vd5} = V_{in}(min) - V_{SS} - \left[\frac{I_5}{\beta_1}\right]^{0.5} - V_{T1}(max)$$

$$= (-3) - (-6) - \left[\frac{4.4\times10^{-6}}{(32.9\times10^{-6})(0.053)}\right]^{1/2} - 0.83 = 0.581$$

Using V_{DS5} calculate $(W/L)_5$

$$(W/L)_5 = \frac{2I_5}{K_5[V_{DS5}(sat)]^2} I = \frac{2(4.4\times10^{-6})}{(32.9\times10^{-6})(0.581^2)} = 0.792$$

6. Find $(W/L)_6$ by letting the second pole (p_2) be equal to 2.2 times GB. Assume that $V_{DS6} = V_{DS6}(sat) = V_{DD}$ - V_{out}(max)

$$g_{m6} = 2.2 g_{m2}(C_L/C_c) = 2.2(2.765\times10^{-6})(2/0.44) = 27.65\mu S$$

$$(W/L)_6 = \frac{g_{m6}}{K_6 V_{DS6}(sat)} = \frac{27.65\times10^{-6}}{(15.3\times10^{-6})(1)} = 1.81$$

7. Calculate I_6

$$I_6 = \frac{g_{m6}^2}{2K_6(W/L)_6} = \frac{(27.65\times10^{-6})^2}{(2)(15.3\times10^6)(1.81)} = 13.8\mu A$$

8. Design $(W/L)_7$ to achieve the desired current ratios between I_5 and I_6

$$(W/L)_7 = (I_6/I_5)(W/L)_5 = 0.792\frac{13.8\times10^{-6}}{4.4\times10^{-6}} = 2.484$$

9. In the final step, find W from W/L ratio. Therefore, we have

$W_1=W_2=4.5\mu m$

$W_3=W_4=4.5\mu m$

$W_5=4.5\mu m$

$W_6=1.81(4.5-0.28\times2)=7.13\mu m$

$W_7=2.484(4.5-0.28\times2)=9.79\mu m$

10. Readjust W_6 for proper balance.

$$2\frac{W_7}{W_5}=\frac{W_6}{W_4} \quad then W_6=10\times2=20\mu m$$

APPENDIX 4

SPICE Code for CMOS Opamp

```
CMOS Op-amp Circuit
.width out=80
vin+ 1 0 0 ac 1 pulse(-2 2 5u 0.1u 0.1u 10u)
vdd 4 0 dc 6
vss 0 5 dc 6
cl 3 0 2p
x1 1 3 3 4 5 opamp
.subckt opamp 1 2 6 8 9
m1 4 2 3 3 nmos1 w=4.5u l=4.5u
m2 5 1 3 3 nmos1 w=4.5u l=4.5u
m3 4 4 8 8 pmos1 w=4.5u l=4.5u
m4 5 4 8 8 pmos1 w=4.5u l=4.5u
m5 3 7 9 9 nmos1 w=4.5u l=4.5u
m6 6 5 8 8 pmos1 w=20u l=4.5u
m7 6 7 9 9 nmos1 w=10u l=4.5u
m8 7 7 9 9 nmos1 w=4.5u l=4.5u
cc 5 6 0.44p
ibias 8 7 4.4u
.model nmos1 nmos level=2 ld=0.28u tox=50n nsub=1e16
+ vto=0.827125 kp=32.86649u gamma=1.35960  phi=0.6 uo=200.00
+ uexp=1.001m ucrit=999000 delta=1.24050 vmax=100000. xj=0.4u
+ lambda=0.01604983 nfs=1.234795e+12 neff=1.001e-2 nss=0. tpg=1.
+ rsh=25 mj=0.5 mjsw=0.33 cgso=520p cgdo=520p cj=320u cjsw=900p
```

```
.model pmos1 pmos level=2 ld=0.28u tox=50n nsub=1.121088e14
+ vto=-0.894654 kp=15.26452u gamma=0.879003  phi=0.6 uo=100.00
+ uexp=0.153441 ucrit=16376.5 delta=1.93831 vmax=100000. xj=0.4u
+ lambda=0.04708659 nfs=8.788617e+11 neff=1.001e-2 nss=0. tpg=-1.
+ rsh=95 mj=0.5 mjsw=0.33 cgso=400p cgdo=400p cj=200u cjsw=450p
.ends
.op
.options limpts=501
.tf v(3) vin+
.dc vin+ -6 6 0.2
.ac dec 10 1 10meg
.trans 0.1u 20u
.print trans v(1) v(3)
.print dc v(3)
.print ac vm(3) vp(3)
.end
```

APPENDIX 5

Diagnosability Measurement

Theorem 6.2

A network is t-diagnosable, $t \geq 2$, if and only if it is $(t-1)$-diagnosable and the columns of matrix $[D_{t-1}, D_t]$ are all distinct.

Lemma 6.1

If the columns of $[D_1, D_2]$ are all distinct, and the columns of $[D_2, D_3]$ are also all distinct, then the columns of $[D_1, D_3]$ are also all distinct.

Proof: Let c_i and c_{jkl} be any columns of D_1 and D_3, respectively, and c_{ij} be a column of D_2. We claim that c_i is distinct to c_{jkl}. Assume that they are identical, i.e., $c_i = c_{jkl}$. This implies that for every element t, $c_i^t = c_{jkl}^t = x$, where x can be either 0 or 1. Two cases can be identified. Case 1, If x=1, then $c_{ij}^t = c_i^t + c_j^t = 1 = x$, i.e. $c_i^t = c_{ij}^t$; and Case 2, if x=0, then $c_{jkl}^t = c_j^t + c_k^t + c_l^t = 0 \Rightarrow c_j^t = 0 = x = c_{ij}^t$. Both two cases conclude that $c_i^t = c_{ij}^t$ for every t, or $c_i = c_{ij}$, which contradicts to the fact that the columns of $[D_1, D_2]$ are all distinct. Therefore, $c_i \neq c_{jkl}$, or columns of $[D_1, D_3]$ are all distinct.

Lemma 6.2

If the columns of $[D_{t-1}, D_t]$ are all distinct and the columns of $[D_i, D_{t-1}]$, for $i < t$, are also all distinct, then the columns of $[D_i, D_t]$ are all distinct.

Proof: The proof is similar to Lemma 6.1.

Proof of Theorem 6.2: If network N is t-diagnosable, by Theorem 6.1, the columns of $D=[D_1,D_2,\cdots D_{t-1},D_t]$ are all distinct. This implies that the columns of $[D_1,\cdots,D_{t-1}]$ are all distinct and the columns of $[D_{t-1},D_t]$ are also all distinct. Again, by Theorem 6.1, the former implies that the network is $(t-1)$-diagnosable.

On the other hand, if the network is $(t-1)$-diagnosable, then columns of $[D_1,\cdots,D_{t-1}]$ are all distinct. This implies that the columns of $[D_i,D_{t-1}]$, for i<t-1, are also all distinct. Further, since the columns of $[D_{t-1},D_t]$ are known to be all distinct, by Lemma 6.2 the columns of $[D_i,D_t]$, for $i<t$ are all also distinct. In the other words, the columns of $D=[D_1,\cdots,D_{t-1},D_t]$ are all distinct. This results that the network is t-diagnosable.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Allen, P.E. and D.R. Holberg, *CMOS Analog Circuit Design*, Holt, Rinehart and Winston, Inc., New York, NY., 1987.

[2] Allen, R. J., "Fault Prediction Employing Continuous Monitoring Techniques," *IEEE Trans. on Aerosp. Support Conf. Procedures*, Vol. AS-1, pp. 924-930, 1963.

[3] Bandler, J. W. and A. E. Salama, "Fault Diagnosis of Analog Circuit," *IEEE Proceedings*, Vol. 73, No. 8, pp.1279-1325, Aug. 1985.

[4] Bennetts, R. G., *Design of Testable Logic Circuits*, Addison-Wesley Publishing Company, 1984.

[5] Berkowitz, R. S., "Conditions for Network-Element-Value Solvability," *IRE Trans. Circuit Theory*, Vol. CT-9, pp. 24-29, Jan. 1962.

[6] Biernacke, R. M. and J. W. Bandler, "Multi-fault Location in Analog Circuits," *IEEE Tans. on Circuits and Systems*, ms, Vol. CAS-28, No. 5, pp.361-3666, May 1981.

[7] Chua, L. O, and P. M. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithm and Computational Techniques*, Prentice-Hall, 1975.

[8] Duhamel, P. and J. C. Rault, "Automatic Test Generation Techniques for Analog Circuits and Systems: A Review," *IEEE Trans. Circuits and Systems*, Vol. CAS-26, pp. 411-440, July 1979.

[9] Greenbaum, J. R., L. Besser, B. Biehl, B. D. Pollard and R. Osann, *Analysis and Design of Electronic Circuits using PCs*, Van Nostrand Reinhold Company, New York, 1988.

[10] Gupta, H., Bandler, J.W., Starzyk, J.A., and J. Sharma, "A Hierarchical Decomposition Approach for Network Analysis," Proc. of IEEE International Symp. on Circuits and Systems, Rome, Italy, pp.643-646, 1982.

[11] Harris Incorporation, *Analog Product Data Book*, 1986.

[12] Hochwald, W. and J. D. Bastian, " A dc Approach for Analog Fault Dictionary Determination," *IEEE Trans. Circuits and Systems*, Vol. CAS-26, pp. 523-529, July 1979.

[13] Huang, T. S. and R .R Parker, "Network Theory: an Introductory Course", Addison-Wesley, 1971.

[14] Huang, Z. F. and R. Liu, "Analog Fault Diagnosis with Tolerance," *Proc. 1986 IEEE Int. Symp. on CAS*, San Jose, CA, pp. 1332-1335, May 1986.

[15] Huang, Z. F. , C. S. Lin and R. Liu, "Node-Fault Diagnosis and a Design of Testability," *IEEE Trans. on Circuits and Systems,* Vol. CAS-30, No. 5, pp. 257-265, May 1983.

[16] Jaeger, R. C., "Tutorial: Analog Data Acquisition Technology, Part II Analog-to-Digital Conversion," *IEEE Micro*, pp. 46-56, Aug. 1982.

[17] Jaeger, R. C., "Tutorial: Analog Data Acquisition Technology, Part III Sample-and-holds, Instrumentation Amplifiers, and Analog Multiplexers," *IEEE Micro*, pp. 20-36, Nov. 1982.

[18] Jaeger, R. C., "Tutorial: Analog Data Acquisition Technology, Part IV System Design, Analysis, and Performance," *IEEE Micro*, pp. 52-60, Feb. 1983.

[19] Jiang B. L. and L. J. Fan, "Multiple-fault Diagnosis of Analog Circuit with Inaccessible Nodes," *Proc. 27th Midwest Symp. on Circuits and Systems*, West Virginia pp. 626-631, June 1984.

[20] Jiang, B. L. and C. L. Wey, "Multiple Fault Diagnosis with Failure Bound for Analog Circuits," *Proc. 1986 IEEE Int. Symp. on Circuits and Systems,* San Jose, CA, pp. 1261-1264, May 1986.

[21] Jiang, B. L., C .L. Wey and L. J. Fan, "Fault Prediction for Analog Circuits," *Journal of Circuits, Systems and Signal Processing*, Vol. 7, No. 1, pp. 95-109, Jan. 1988.

[22] Jiang, B. L. and C. L. Wey, "Fault Prediction Precess for Large Analog Circuit Networks," *International Journal of Circuit Theory and Applications*, Vol. 17, No. 2, pp. 141-149, April 1989.

[23] Jiang, B. L., *AATPG User Manual*, Dept. of Electrical Engineering, Michigan State University, 1989.

[24] Liguori, F., Editor, *Automatic test Equipment: Hardware, Software, and Management*,, IEEE Press, New York, 1974.

[25] Liu, R., *Analog Fault Diagnosis*, IEEE, PRESS, 1987.

[26] Liu, R., C. S. Lin, Z.F. Huang and L. Z. Hu, "Analog Fault Diagnosis: a New Circuit Theory," *Proc. 1983 IEEE int. Symp. Circuits and Systems*, Newport Beach, CA, pp. 931-939, May 1983.

[27] McCluskey, E. J., "Built-In Self-Test Technique," *IEEE Design and Test of Computers*, Vol. 2, No.2, pp. 21-28, April 1985.

[28] McCluskey, E. J., "Built-In Self-Test Structures," *IEEE Design and Test of Computers*, Vol.2, No.2, pp. 29-36, April 1985.

[29] Milor, L. and V. Visvanathan, "Efficient GO/NO-GO Testing for Analog Circuits," *Proc. of IEEE 1987 Int. Symp. on Circuits and Systems*, Philadephia, PA, pp. 414-417, May 1987.

[30] Nagel, L. W., "SPICE2: A Computer Program to Simulate Semiconductor Circuits," Univ. of California, Berkeley, 1975.

[31] Ozawa, T., J.W. Bandler, and A. E. Salama, "Diagnosability in the Decomposition Approach for Fault Location in Large Analog Network," *IEEE Trans. on Circuits and Systems*, Vol. CAS-32, No. 4, pp. 415-416, April 1984.

[32] Plice, W. A., "Automatic Analog Testing -1979 Style," *Computer*, pp. 40-47, Oct. 1979.

[33] Plice, W.A., "Overview of Current Automated Analog Test Design," *IEEE Test Conference*, Cherry Hill, New Jersey, pp. 128-136, 1979.

[34] Rapisarda, L. and R. DeCarlo, "Analog Multifrequency Fault Diagnosis," *IEEE Trans. on Circuits and Systems*, Vol. CAS-30, pp. 223-234, 1983.

[35] Reisig, D. and R. DeCarlo, "A Method of Analog-Digital Multiple Fault Diagnosis," *Int. Journal of Circuit Theory an Applications*, Vol. 15, pp. 1-22, April, 1987.

[36] Saeks, R., "Criteria for Analog Fault Diagnosis," in *Nonlinear Fault Analysis*, Technical report. Texas Tech U., Lubbock, Texas, pp.19-28.

[37] Saeks. R., "An Experiment in Fault Prediction," *Proc. 1976 Autotestcon*, Arlington, TX., pp. 53, Nov. 1976.

[38] Saeks. R. , and S. R. Liberty, *Rational Fault Analysis* Marcel Dekker, 1977.

[39] Salama, A. E., J. A. Starzyk and J. W. Bandler, "A Unified Decomposition Approach for Fault Location in Large Analog Circuits," *IEEE Trans. on Circuits and Systems*, Vol. CAS-31, No. 7, pp. 609-622, July 1984.

[40] Sen, N. and R. Saeks, "Fault Diagnosis for Linear System via Multifrequency Measurements," *IEEE Trans. on Circuits and Systems*, Vol. CAS-26, pp. 457-465, 1979.

[41] Siliconix Incorporated, *Integrated Circuit Data Book*, 1988.

[42] Starzyk, J. A. and H. Dai, "Multifrequency Measurement of Testability in Analog Circuits," *Proc. 1987 IEEE int. Symp. Circuits and Systems*, Philadelphia, PA, pp. 884-887, 1987.

[43] Trick T, N. , W. Mayeda, and A. A. Sakla, "Calculation of Parameter Values from Node Voltage Measurements," *IEEE Trans. Circuits and Systems*, Vol. CAS-26, pp. 440-457, 1979.

[44] Tsui, F.F., *LSI/VLSI Testability Design*, McGraw-Hill, 1986.

[45] Wey, C. L., "Design for Testability for Analogue Fault Diagnosis," *International Journal of Circuit Theory and Applications*, Vol. 15 , pp. 123-142, April, 1987.

[46] Wey, C.L., B.L. Jiang and G. Wierzba, "Built-In Self-Test for Analog Circuit Networks," *Proc. 31th Midwest Symp. on Circuits and Systems*, St. Louis, MO. August 1988.

[47] Wey, C.L., B.L. Jiang and G. Wierzba, "Built-In Self-Test (BIST) Design of Large-Scale Analog Circuit Networks," *Proc. IEEE Int. Symp. on Circuits and Systems*, Portland, OR. May 1989.

[48] Wey, C. L. and R. Saeks, "On the Implementation of Analog ATPG: The Linear Case," *IEEE Trans. on Instrumentation and Measurerment*, Vol. IM-34, pp. 442-449, September, 1985.

[49] Wey, C. L. and R. Saeks, "On the Implementation of Analog ATPG: The Nonlinear Case," *IEEE Trans. on Instrumentation and Measurerment,* Vol. IM, pp. 442-449, June, 1988.

[50] Wu, C.-c., K. Nakajima, C. L. Wey, and R. Saeks, "Analog Fault Diagnosis with Failure Bounds," *IEEE Trans. on Circuits and Systems*, Vol. CAS-29, No.5, pp. 277-284, May, 1982.