THE UNIVERSITY OF ALBERTA

FAULT DETECTION IN COMBINATIONAL NETWORKS

by

(C) MAREK FRIDRICH

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTING SCIENCE

EDMONTON, ALBERTA

SPRING, 1973

ABSTRACT

This thesis is concerned with the problem of minimal tests to detect stuck-at-0 and stuck-at-1 faults in combinational networks. A procedure that generates a minimal single fault test set for any irredundant network is given. Fault masking is studied and a method is provided to find multiple faults that are not detected by a test set for single faults. Sufficient conditions for multiple fault detection are derived and procedures to obtain both a minimal and a nearly minimal multiple fault test set are described. The problem of faults in redundant networks is examined and it is shown how to obtain an optimal test set for multiple fault detection. A lower and an upper bound on the number of tests required to detect all single faults in an irredundant network is given. Finally, the problem of fault diagnosis is briefly examined and methods for obtaining the diagncstic test set are proposed.

ACKNOWLEDGEMENT

The author wishes to express sincere appreciation to Dr. Wayne A. Davis, his supervisor, for guidance and continued encouragement throughout the period of research for this thesis. The support extended by the Department of Computing Science and the National Research Council under Grant A7634 is appreciated. The author is also grateful to his wife, Marie, for her patience and understanding during the past years. The assistance of Mrs. Marilyn Wahl in typing of the manuscript is gratefully acknowledged.

TABLE OF CONTENTS

CHAPTER PAGE

I. INTRODUCTION 1

1.1 Assumptions and Definitions 2

1.2 Previous Results 7

1.3 Outline of Current Results 9

II. SINGLE FAULT DETECTION IN IRREDUNDANT NETWORKS 12

2.1 The Equivalent Sum of Products Form . . 15

2.2 The Single Fault Procedure 26

III. MULTIPLE FAULT DETECTION IN IRREDUNDANT NETWORKS 33

3.1 Fault Masking and Minimal Test Sets for Multiple Faults 35

3.2 Sufficient Conditions for Multiple Fault Detection 52

IV. FAULT DETECTION IN REDUNDANT NETWORKS . . . 60

4.1 Single Fault Detection 61

4.2 Multiple Fault Detection 66

4.3 Concluding Remarks 79

V. SPECIAL NETWORKS 80

5.1 Networks with Equal Inversion Parity of all Reconverging Paths 80

5.1.1 Single Faults 81

5.1.2 Multiple Faults 87

5.2 Multiple Faults in Fanout-free Networks 90

CHAPTER		PAGE
	5.3 Test Generation for Large Circuits . . .	95
	5.4 Multiple Output Networks	99
VI.	SOME BOUNDS ON THE LENGTH OF A MINIMAL TEST SET	106
VII.	FAULT DIAGNOSIS	116
	7.1 Fanout-free Networks	117
	7.2 Networks with Reconvergent Fanout . . .	123
VIII.	CONCLUSIONS	132
REFERENCES	. . . .	135
APPENDIX A	. . . .	142
APPENDIX B	. . . .	144

LIST OF TABLES

TABLE		PAGE
2.1	Karnaugh Map for Network N2.1	30
2.2	Testing Table for Network N2.1	30
2.3	Checkpoint Covering Table for Network N2.1 .	31
3.1	Test Set T_{s1} for Network N3.1	43
3.2	Testing Table for Network N3.1	44
4.1	Testing Table for Network N4.3	70
4.2	Checkpoint Covering Table for Network N4.3 .	71
4.3	Checkpoint Covering Table for Network N4.6 .	76
5.1	Testing Table for Network N5.1	86
5.2	Checkpoint Covering Table for Network N5.1 .	86
5.3	Testing Table for Network N5.5	102
5.4	Checkpoint Covering Table for Network N5.5 .	104
7.1	Fault Dictionary for Network N7.2	122
7.2	Test Set T_1 for Network N7.4	127
7.3	Fault Dictionary for Network N7.4	130

LIST OF FIGURES

FIGURE		PAGE
1.1	Logical gates	4
1.2	Exclusive-OR modules	5
2.1	Network N2.1	13
2.2	Network N2.2	22
3.1	Network N3.1	42
3.2	Masking graph for network N3.1	42
3.3	Masking graph for network N2.1	49
3.4	Network N3.4	51
3.5	Network N3.5	57
4.1	Network N4.1	62
4.2	Network N4.2	64
4.3	Network N4.3	69
4.4	Masking graph for network N4.3	69
4.5	An example of a masking graph	73
4.6	Network N4.6	77
5.1	Network N5.1	85
5.2	Network N5.2	89
5.3	Network N5.3	91
5.4	Network N5.4	97
5.5	Network N5.5	101

FIGURE PAGE

6.1 Network N6.1 111

6.2 Network N6.2 111

6.3 Network N6.3 113

6.4 Network N6.4 113

7.1 Network N7.1 118

7.2 Network N7.2 121

7.3 Network N7.3 125

7.4 Network N7.4 128

Chapter 1

INTRODUCTION

This thesis deals with the problem of tests for detecting and locating faults in combinational networks. To ensure the correct operation of a digital system, the system must be tested periodically. If a failure is detected, the faulty component must be identified and replaced. The widening use of digital circuits together with the increased complexity of the hardware being built have emphasized the need for efficient testing procedures. The necessity to protect the system against failures is particularly strong with computers operating in a real-time environment. Although fault masking and different types of redundancy schemes can be used to combat failures, these techniques do not eliminate the need of efficient testing procedures since they are only capable of prolonging the time interval between testing.

1.1 Assumptions and Definitions

The faults considered herein are faults that produce a change in the logical behaviour of the circuit. Such logical faults can be either permanent or intermittent (i.e. transient). Although intermittent faults are important, only permanent faults will be dealt with. Very little has been said with respect to designing tests for transient faults since such faults may disappear by the time a test is applied.

Most permanent faults in logical circuits can be represented [32] by an input or the output of some gate being stuck-at-1 (denoted s-1) or stuck-at-0 (s-0). A single fault occurs when only one connection in the network is s-0 or s-1. A multiple fault consists of two or more single faults occuring simultaneously. A test for a particular fault is an input combination such that the output of the correct network is 1 and the output in the presence of the fault is 0, or vice versa. A test set is a set of one or more tests. A test set that detects all faults of a given type is said to be a complete test set. Since a test set that is not complete is of little use, "test set" will be used to denote a "complete test set" unless otherwise stated. For any combinational network the set of all possible input combinations represents a trivial example of a test set which is complete. Because of its size such a test set is only used for very simple circuits. The <u>length of a test set</u> is the number of tests in the set. A complete test set T for a given combinational network is said to be a <u>minimal test set</u> if there does not exist a test set T' such that the length of T' is smaller than the length of T.

Since each input or output line of an n-input gate may be s-0 or s-1, there are 2(n+1) single faults and $3^{n+1}-1-2(n+1)$ multiple faults that may change the switching function realized by a single gate. For any n-input OR, AND, NOR, NAND or NOT gate, (n+1) input combinations are sufficient to detect all faults on the inputs and output of the gate; see Eldred [13] and Schertz [44]. For example, the test set ($\bar{a}\bar{b}\bar{c}$, $a\bar{b}\bar{c}$, $\bar{a}b\bar{c}$, $\bar{a}\bar{b}c$) detects all s-0 and s-1 faults on input and output lines of the OR gate in fig. 1.1a), or the NOR gate in fig. 1.1b). Similarly, the test set (abc, $\bar{a}bc$, $a\bar{b}c$, $ab\bar{c}$) is a complete test set for the AND gate in fig. 1.1c) or the NAND gate in fig. 1.1d). Unless otherwise stated, it is assumed that the networks considered are represented by diagrams where only AND, OR, NAND, NOR or NOT gates appear. The reason for this restriction is that the s-0 or s-1 fault model is not sufficient to properly describe the faulty properties of the more complex logical modules. For example, the test set (ab, $\bar{a}b$, $a\bar{b}$) detects all s-0 and s-1 faults on input and output lines of the Exclusive-OR module in fig. 1.2a). A practical realization of such a module is shown in fig. 1.2b). If input

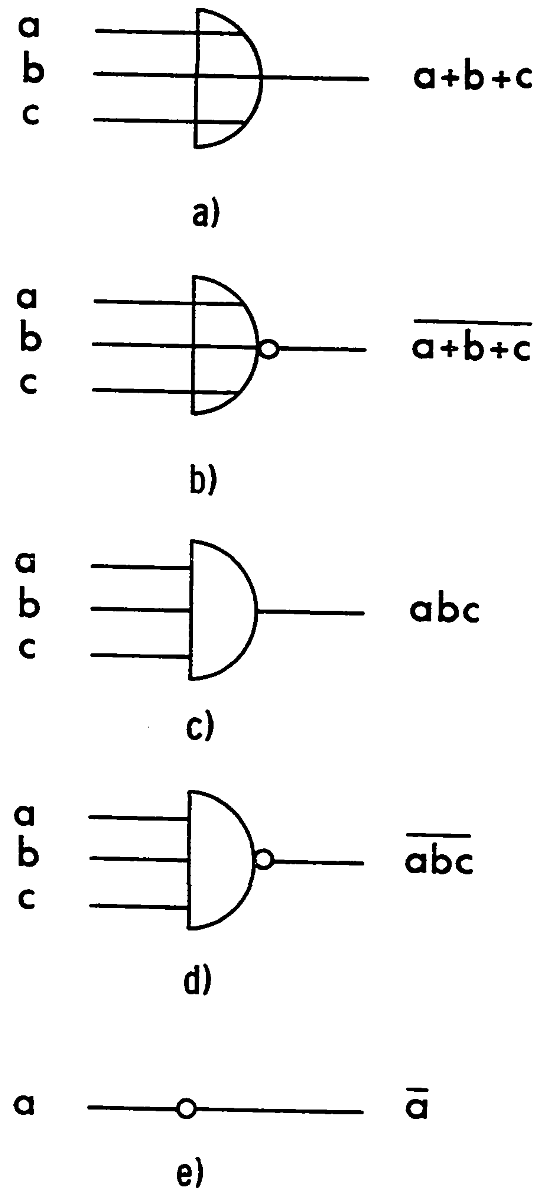
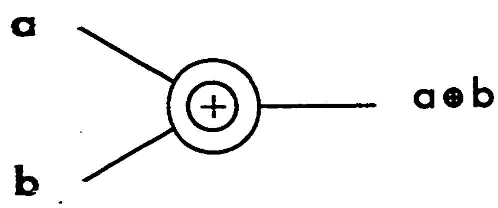
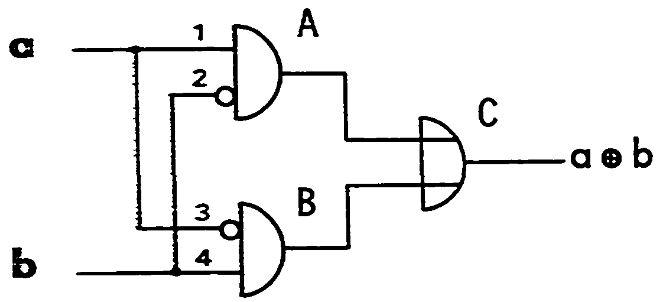


Figure 1.1 Logical gates



a)



b)

Figure 1.2 Exclusive-OR modules

1 to gate A is open (this corresponds to the fault "line 1 is s-1"), then the function realised in the presence of this fault is $\bar{b}+\bar{a}b$, but the above test set does not detect the fault.

A minimal test set for any combinational circuit can always be selected from the <u>fault table</u> (or <u>matrix</u>) of the network [25,35]. If n is the number of input variables and m is the number of possible faults in the network, then this table has 2^n rows and m+1 columns. The first column of the table represents the output of the correct network under each of the 2^n possible input combinations. The remaining m columns represent the output functions realised by the network in the presence of the m possible faults. By using this table, the problem of finding a minimal test set is equivalent to the problem of selecting a minimal cover [35]; however, due to the size of the fault table, the above method is practical only in a limited number of cases.

The derivation of tests is greatly simplified when it is assumed that only single faults occur. Using this assumption, a number of methods for test set derivation have been developed. Considerably less attention has been given to the problem of multiple faults and to the problem of fault detection in redundant networks. In redundant networks certain faults are not detectable. This fact complicates the derivation of the test set, specifically when multiple faults are considered.

1.2 Previous Results

Present methods [9,16] for fault detection in combinational circuits usually generate a single fault test set that is nearly minimal. Until recently [5,12,29], the problem of multiple fault detection has not been studied extensively .

One of the earliest methods to generate tests for irredundant circuits was described by Poage [35]. This method derives literal propositions that describe the output function and also represent the structure of the network. These propositionsare then employed in generating the test set. Although producing a minimal test set, this method is very laborous even for simple networks.

Armstrong [2] has shown that a test that detects a fault in the circuit must form a sensitized path from the fault tested to the primary output. This method generates a nearly minimal test set with a medium amount of computational effort.

The path sensitizing concept was utilized by several authors [10,21,38]. A well known method is the calculus of D-cubes developed by Roth [41,42]. The D-algorithm mechanizes the process of forming sensitized paths and also guarantees that a test is generated for every detectable fault. There is no guarantee, however, that the test set generated will be a minimal test set.

A minimal test set for a two level network can be generated by using the prime implicants of the function realized by the network [28]. Paige [33] has employed prime implicants to derive tests for irredundant multiple level networks with certain structures. Bearnson and Carroll [4] obtained similar results by using Boolean differences. The big advantage of the above methods is that they require a smaller amount of computation than the path sensitizing methods or the analytical method of Poage [35]; however, they are not applicable to an arbitrary network.

Schertz [44] studied the indistinguishability of certain faults and has shown how to combine such indistinguishable faults into equivalence classes. He also defined the class of restricted fanout-free networks. For such networks any test set that detects all single faults also detects all multiple faults.

Recently, the first general solution to the problem of multiple faults was given by Bossen and Hong [5]. This method (procedure G in [5]) is applicable to irredundant as well as redundant combinational networks, but it may produce a test set that is far from being optimal. Procedure NR[5], which is claimed to generate a nearly minimal test set for any irredundant network, does not guarantee detection of all faults in networks with reconvergent fanout. Kohavi and Kohavi [29] have shown how to generate a nearly minimal multiple fault test set for networks with irredundant

Equivalent Sum of Products form.

Ramamoorthy [40] has considered the structural properties of large systems represented by system graphs and has derived some theoretical criteria for good diagnosibility.

The problem of fault detection in cellular arrays has been studied by Kautz [24] and Friedman and Mennon [16].

1.3 Outline of Current Results

The main objective of this study is to provide methods that would generate a minimal or a nearly minimal test set for the detection of single as well as multiple faults in combinational networks.

Detection of single faults in irredundant networks is treated in chapter 2. It is shown that in order to detect all faults in the network it is sufficient to detect all faults on input lines and fanout branches (called checkpoints). The Equivalent Sum of Products form (denoted ESP form) of a combinational network is defined and the effect of faults upon the terms of this form is examined. Finally, a procedure that generates a minimal single fault test set is given.

The multiple faults not detected by a single fault test set consist of a number of faults that prevent detection of each other. In chapter 3, the conditions under which fault masking occurs are stated and proved. It is shown how to find multiple faults that are not detected by a single fault test set. If such undetected multiple faults exist, then the multiple fault test set is obtained by enlarging the single fault test set previously derived. Some sufficient conditions for multiple fault detection are given and it is demonstrated how these conditions can be used to generate a nearly minimal multiple fault test set with considerable saving of computational effort.

Chapter 4 deals with the problem of fault detection in redundant networks. These networks are further complicated by the fact that the detection of all detectable single faults on checkpoints is not sufficient to detect all single faults that are detectable. It is shown, however, that a complete test set can be derived without considering faults on all connections in the network. A multiple fault that occurs in a redundant circuit may consist of detectable single faults, or undetectable single faults, or both. Multiple faults belonging to these three classes are analysed, and a procedure that generates a minimal multiple fault test set for any redundant network is given.

The generation of tests is much easier for networks where the inversion parity of all reconverging paths is the same. A simple method that derives a minimal single fault test set for such networks is described in chapter 5. It is shown that any single fault test set for a fanout-free network detects all multiple faults of multiplicity two and three. An alternative way of generating tests, which may be suitable for large networks, is also described.

In chapter 6 some bounds on the length of a minimal test set are given. It is shown that at least $2\sqrt{p}$ and at most $\frac{3}{2}p$ tests are required to detect all single faults in a fanout-free network with p checkpoints, where $p > 1$. Similar bounds are also derived for networks with fanout.

The problem of fault location is examined briefly in chapter 7, and two methods for obtaining the diagnostic test set are suggested.

Chapter 2

SINGLE FAULT DETECTION IN IRREDUNDANT NETWORKS

A network is usually considered to be redundant whenever parts of the network can be removed without changing the output function. The following definition is a direct consequence.

Definition 2.1: A combinational network is irredundant if it is possible to detect all permanent s-0 and s-1 faults within the network.

To consider the set of all possible single faults that can occur in a network would yield an incredibly complex procedure for finding tests. It is therefore important to reduce this set of faults to a subset occuring at specific places in the network.

Definition 2.2: The checkpoints of a combinational network are [5]

(a) all primary inputs that do not fanout and

(b) all fanout branches.

To illustrate, consider the network in fig.2.1. The primary inputs are enumerated as checkpoints 1 to 4 while the fanout branches are 5 thru 8.

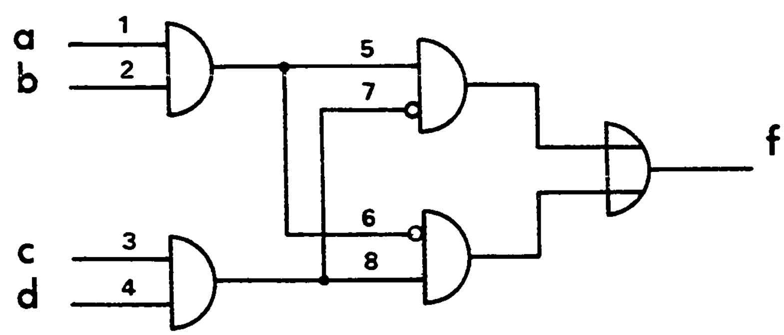


Figure @.1 Network N2.1

It will be now shown, that in order to detect all single faults in an irredundant network, it is sufficient to detect only the single faults on the checkpoints.

Lemma 2.1: [29] All single faults in an irredundant combinational network are detected if and only if all single faults on the checkpoints are detected.

Proof: Whatever method is used for the generation of the test set, in the actual circuit there must always be a sensitized path (or paths) [2] from the fault tested to the primary output of the network. Each connection belonging to this path must carry a specific value (0 or 1) when the fault is absent and the opposite value when the fault is present. Any fanout-free network has a tree structure and there is always a unique path from any input terminal to the primary output of the network. If s-0 and s-1 faults on all the input terminals are detected, then the sensitized paths include all wires in the fanout-free network and consequently all single faults within the network are detected as well.

A network containing fanout can be decomposed into a set of subnetworks each being fanout-free and having all inputs that are checkpoints. Therefore the result for fanout-free networks holds, provided that a multiple fault produced by a single fault in a subnetwork feeding a second net is detected by the tests for single faults on the second network. Without loss of generality, consider the case of two subnetworks where the output of the first subnetwork fans out to provide two inputs to the second subnetwork. Suppose that none of the tests that detect a single fault on these two inputs will detect the multiple fault on both inputs. This can only occur if there does not exist an input condition on the remaining inputs such that the paths from neither input are allowed to propogate to the output whenever both inputs are identical. If this is the case then the output is not a function of the two inputs since the inputs are always the same in the fault-free network. This is a contradiction which completes the proof of the lemma.

This result simplifies the problem considerably. In order to find a minimal single fault test set for the given network, a minimal test set detecting only those faults on the checkpoints must be found.

2.1 The Equivalent Sum of Products Form

The test set for a given network depends on the structure of the network as well as the Boolean function realized by the network. Therefore both the structure and the output function must be considered in the process of test generation. A relatively simple expression describing the output function of the network while preserving the important structural properties is the Equivalent Sum of Products form of the network. Because this form (or similar equivalents) has been used already by some authors [2, 35, 5] it is introduced here only briefly by definition and simple example.

Definition 2.3: The Equivalent Sum of Products form (denoted ESP(f)) of a network is obtained from the Boolean expression describing the network by:

(a) for each checkpoint in the network associate a subscript in the corresponding position in the expression

(b) expand the expression collecting subscripts, from fanout points, onto the input literals such that the subscripts attached to input variables in the resulting sum of products expression denote the propagating path of that particular input variable

(c) do not discard redundant terms or literals.

For the network in fig. 2.1 the ESP(f) form is obtained by expanding the expression

$$f = (a_1 b_2)_5 \, (\overline{c_3 d_4})_7 + (\overline{a_1 b_2})_6 \, (c_3 d_4)_8$$

$$ESP(f) = a_{15} b_{25} \bar{c}_{37} + a_{15} b_{25} \bar{d}_{47} + \bar{a}_{16} c_{38} d_{48} + \bar{b}_{26} c_{38} d_{48}$$

With regard to notation, it should be pointed out that if there are 10 or less checkpoints then the notation used above is unambiguous since the digits 0 thru 9 can be used; however if there are more than 10 it will be necessary to insert commas between the numbers and the following notation will be used:

$$ESP(f) = a_{1,5}b_{2,5}\bar{c}_{3,7}+a_{1,5}b_{2,5}\bar{d}_{4,7}+\bar{a}_{1,6}c_{3,8}d_{4,8}+\bar{b}_{2,6}c_{3,8}d_{4,8}$$

The terms of the ESP(f) form represent the set of all conditions for which the output of the network is 1. Similarly, the set of all conditions for which the output of the network 0 is given by the complemented form ESP($\bar{f}$). The ESP($\bar{f}$) form is obtained by first complementing the output expression and subsequent expansion. For the network in fig. 2.1.

$$\bar{f} = \overline{(a_1b_2)_5 \; (\overline{c_3d_4})_7 + (\overline{a_1b_2})_6 \; (c_3d_4)_8}$$

$$ESP(\bar{f}) = \bar{a}_{15}a_{16}b_{26} + \bar{a}_{15}\bar{c}_{38} + \bar{a}_{15}\bar{d}_{48} + a_{16}b_{26}\bar{b}_{25} + \bar{b}_{25}\bar{d}_{48} +$$
$$+ \bar{b}_{25}\bar{c}_{38} + a_{16}b_{26}c_{37}d_{47} + \bar{c}_{38}c_{37}d_{47} + c_{37}d_{47}\bar{d}_{48}$$

The notation ESP will be used for ESP(f) or ESP($\bar{f}$) where no differentiation is necessary or when implying both.

Definition 2.4: The ESP form is irredundant if it contains no redundant terms or literals.

The ESP form is obtained by expanding the Boolean expression describing the output function of the network. This expression is given by the type and connections to the gates in the network. As a result the ESP form is a function of the network structure. When dealing with the two ESP forms the following notation will be used:

$$ESP(f) = \Sigma X_i, \text{ and } ESP(\bar{f}) = \Sigma Y_j.$$

In an irredundant network the occurrence of any single fault affects at least one X_i and at least one Y_j term. For the above example, the s-1 fault on checkpoint 2 (denoted by 2-1) would cause the term $X_2 = a_{15}b_{25}\bar{d}_{47}$ to become independent of variable b and X_2 would grow into $X_2' = a_{15}\bar{d}_{47}$; 2-1 would also cause the growth of X_1 and Y_7, and the disappearance of terms X_4, Y_5 and Y_6. The effect of a particular fault upon a given X_i or Y_j term will be established formally by using the definition of a normal network. A <u>normal</u> network is a network having no complemented input variables. Clearly, any given network can be transformed into a normal network by adding inverters to the primary input lines where necessary. Because of the inability to distinguish between a fault at the input of an inverter and the complemented fault at the output, any test set detecting all faults in the original network also detects all faults in the associated normal network and vice versa. In order to simplify the discussion, only normal networks will be dealt with.

Let $a_{pq...s}$ W be a term of the ESP form of a given normal network. The fault p-1 will cause the term to become independent of variable a, since the value on checkpoint p is 1 irregardless of the value of the applied variable. By the same reasoning, the fault p-0 would cause the disappearnace of the term $a_{pq...s}$ W. The subscripts associated with each literal denote the path through which the effect of a particular variable propogates to the output. If a fault occurs along this path, then such a fault will cause either the disappearance or growth of the given term, depending on the coincidence, or the discrepancy, between the value on that line in a fault-free circuit and the value imposed by the fault. Consequently, a fault k-1 (or k-0), where $k \varepsilon p,q,\ldots,s$, will cause the term $a_{pq\ldots s}W$ to become independent of variable a if the value at k in the fault-free circuit under input aW is 1 (0). The value at k in the fault-free circuit can be determined from the inversion parity along the path p,q,...,k. For the term $a_{pq\ldots s}W$, the value at k under input aW is 1 (0), if the inversion parity along the path p,q,...,k is even (odd). The dual situation applies if the literal is primed for the term $\bar{a}_{pq\ldots s}W$ grows due to fault p-0 and it disappears due to fault p-1. To summarize, by using the definition of a normal network the effect of a fault upon a particular term can be determined from the literal (primed or unprimed) and the inversion parity along the particular path.

The effect of faults on X_i and Y_j terms suggests, that the complete test set could be derived by checking for the presence and growth of the X_i and Y_j terms. In definitions 5 and 6 following, the notation $t \varepsilon P.Q$ means that the test t is contained in the intersection P.Q.

<u>Definition 2.5</u>: A test t checks for the <u>presence</u> of the term $X_\alpha \neq 0$, if $t \varepsilon X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i$.

<u>Definition 2.6</u>: A test t checks for <u>growth</u> of the term $X_i = aW$ (or $Y_j = aW$) due to literal a, if $t \varepsilon W.\bar{f}$ (or $t \varepsilon W.f$), where f and $\bar{f}$ is the output function and its complement, respectively.

There is a difference in how a test which checks for presence and a test which checks for growth of some X_i or Y_j can be used. It is possible to derive a test set for a special class of networks by generating tests which check for the presence and growth of X_i (or Y_j) terms only, and consequently use only one of the two ESP forms. It is difficult, however, to do so for the following cases:

1. The network has reconvergent fanout. The previous discussion suggests, that a test checking for growth of the term aW due to literal a should detect all faults which make the term indpendent of variable a, and that a test checking for the presence of some term should detect all faults which cause the disappearance of the same term. This is true in the first case (the only exception is defined in lemma 3), but it is not true in the second one when the network contains reconvergent fanout. Reconvergent fanout within the network may cause what is called reconvergent fanout cancellation [2]. This happens when the inversion

parity along all the reconverging paths is not the same. The inversion parity of a reconverging path is defined to be the number of inversions, modulo 2, along the path between the specified fanout node and the node of reconvergence. The network in fig. 2.2 illustrates the difficulties caused by reconvergent fanout. For this network the ESP(f) form is

$$ESP(f) = a_{15}b_{25}c_3 + \bar{a}_{16}d_4 + \bar{b}_{26}d_4$$

Faults 1-0 and 2-0 cause disappearnace of $a_{15}b_{25}c_3$. Both the tests t_1 = abcd and $t_2 = abc\bar{d}$ check for the presence of $X_1 = a_{15}b_{25}c_3$; however, only t_2 detects s-0 faults on checkpoints 1 and 2. Test t_1 does not detect the above mentioned faults because it sensitizes two paths with different inversion parity. This corresponds to the fact, that although fault 1-0 causes the disappearance of X_1, it also causes growth of $X_2 = \bar{a}_{16}d_4$ into $X_2' = d_4$. Since $t_1 \varepsilon X_2'$, t_1 does not detect the fault.

2. The ESP form of the network is redundant. If the ESP form is redundant, it is impossible to check for the presence of some X_α such that $X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i = \emptyset$. This fact complicates the search for a minimal test set.

3. Multiple fault analysis. The problem of deriving a minimal multiple fault test set is difficult when only one of the two ESP forms is used. It will be shown in chapter 3, that by using both the ESP(f)

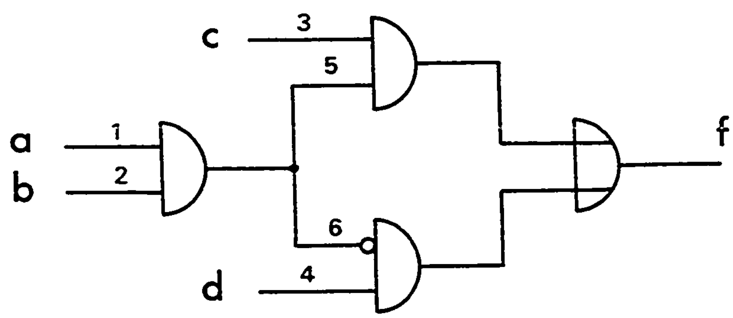


Figure 2.2 Network N2.2

and ESP($\bar{f}$) forms the multiple fault analysis is greatly simplified.

For the reasons stated above, the approach based on checking for growth of both X_i and Y_j terms must be used if fault analysis for the general class of networks is desired. The following lemma defines the faults detected by a test checking for growth of some nonzero term, where term means either a X_i or Y_j term.

Lemma 2.2: Let N be a given normal network with ESP(f) = ΣX_i and ESP($\bar{f}$) = ΣY_j. If a test t checks for the growth of a nonzero term $a_{pq\ldots s}W$ due to literal a, then t detects the following faults on all checkpoints associated with literal a:

-s-1 fault on checkpoint p,

-s-1 fault on checkpoint k ε p,q,...,s if the inversion parity along the path p,q,...,k is even,

-s-0 fault on checkpoint k ε p,q,...,s if the inversion parity along the path p,q,...,k is odd.

If a is replaced by $\bar{a}$ then s-1 faults become s-0 faults and vice versa.

Proof: If t checks for growth of $X_\alpha = a_{pq\ldots s}W$ due to literal a, then t ε $W.\bar{f}$ = $W\bar{a}\bar{f}$ and the output of the network under t is 0, unless there is an s-1 fault on checkpoint p (input terminal). It follows from the way the ESP form is obtained that the output would be also changed if some line k along the path p,q,...,s is s-1 (no inverters or even number of inverters between p and k) or s-0 (odd inversion parity).

For the network in fig. 2.1 the test $abc\bar{d}$ checks for the growth of the term $Y_7 = a_{16}b_{26}c_{37}d_{47}$ due to literal d, since $abc\bar{d} \in a_{16}b_{26}c_{37}.f = abc\bar{d}$; because the network in fig. 2.1 is a normal network, test $abc\bar{d}$ detects fault 4-1 and fault 7-1 (the inversion parity between checkpoints 4 and 7 is even).

It should be noted that for the purpose of the Lemma 2 and Lemma 3 NOR and NAND gates are considered as OR and AND gates, respectively, followed by an inverter. Lemma 2 does not define faults detected by a test checking for growth of zero term. Although terms of the form $a_p\bar{a}_qW$, where $p \neq q$, do not pose a problem, care must be taken when terms of the form $a_{pq}\bar{a}_{pr}W$ are dealt with.

<u>Lemma 2.3</u>: If test t checks for the growth of a term $a_{ij\ldots k}\bar{a}_{pq\ldots r}W$ due to literal a, then t detects faults on the following checkpoints:

(a) on any checkpoint x such that $x \in i,j,\ldots,k$ and $x \notin p,q,\ldots,r$

(b) on any checkpoint x such that $x \in i,j,\ldots,k$ and $x \in p,q,\ldots,r$ and such that the inversion parity along the path $i,j,\ldots,x$ and $p,q,\ldots,x$ is not the same.

The type of faults detected is determined as described in Lemma 2.

Proof: If t checks for growth of $X_\alpha = a_{ij\ldots k}\bar{a}_{pq\ldots r}W$ due to literal a, then $t \in \bar{a}W.\bar{f}$ and will detect all faults which:

1. Cause X_α to become independent of variable a,

and

2. Do not make the term $\bar{a}W$ disappear

This corresponds directly to conditions (a) and (b) stated above.

For example, for the network in fig. 2.2,

$ESP(\bar{f}) = \bar{a}_{15}\bar{d}_4+\bar{b}_{25}\bar{d}_4 + \bar{c}_3\bar{d}_4 + a_{16}b_{26}\bar{c}_3 + \bar{a}_{15}a_{16}b_{26} + a_{16}b_{26}\bar{b}_{25}$

The test $\bar{a}bcd$ checks for growth of $Y_5 = \bar{a}_{15}a_{16}b_{26}$ due to literal a. This literal is associated with checkpoints 1 and 6. The input combination $\bar{a}bcd$ detects fault 6-1, but does not detect fault 1-1, because this fault causes term Y_5 to disappear.

Lemma 2.4: If a test t checks for growth of a term $a_\alpha b_\beta \ldots e_\gamma W$ due to literals a, b,... and e, then t detects faults on those checkpoints which are common to literals a, b,... and e.

Proof: If t checks for growth of X_i = ab...eW due to literals a, b,... and e, then $t \in W.\bar{f}$ and it will detect those faults which make the term X_i independent of all the variables a, b,...,e. Such faults occur on checkpoints common to all the literals.

For example, for the network in fig. 2.1 test $\bar{a}\bar{b}cd$ checks for growth of $Y_7 = a_{16}b_{26}c_{37}d_{47}$ due to literals a and b. It detects fault 6-1, because this fault would cause growth of Y_7 into $Y_7' = c_{37}d_{47}$, and $\bar{a}\bar{b}cd \in Y_7'.f$. Fault 1-1 would cause growth of Y_7 into $Y_7' = b_{26}c_{37}d_{47}$. Test $\bar{a}\bar{b}cd$ does not detect this fault, since $\bar{a}\bar{b}cd$ is not contained in the intersection $bcd.f = \bar{a}bcd$.

In this section we have established the effect of faults upon the terms of the two ESP forms and defined the faults detected by a test that checks for growth of some term. A method that derives a complete test set will now be described.

2.2 The Single Fault Procedure

The following procedure generates a minimal single fault test set for an irredundant combinational network.

Procedure 2.1:

1. Evaluate the $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$ forms of the given network.

2. For each X_i and Y_j generate all terms X_i' and Y_j' which are one variable less. For each X_i' compute $Z_k = X_i'.\bar{f}$. For each Y_j' compute $W_m = Y_j'.f$.

3. If there exists some X_i or Y_j term having more than one Z_k or W_m term that is zero, then repeat step 2 with those X_i' or Y_j' terms having common checkpoints, by removing the variables with common checkpoints according to Lemma 4.

4. Choose a minimal cover MC_1 covering all the Z_k and W_m terms. MC_1 can be obtained by using the method of pairwise intersections [28].

5. Construct the checkpoint covering table where the column headings are the checkpoints of the network and the row headings are the members of $\{MC_1\}$. Entry (i,j) is 1 (0) if test i detects s-1 (s-0) fault on checkpoint j. For the Z_k and W_m terms obtained in step 2 the entries are filled according to Lemma 2 and 3, for those obtained in step 3 according to Lemma 4.

6. Find a minimal cover MC_2 such that all checkpoints are checked for s-0 and s-1 faults. MC_2 is a minimal single fault test set T_s.

Validity of the Procedure: In order to establish the validity of the procedure it is to be proved that the procedure generates a test for every single fault in any irredundant network, and show that the test set generated is a minimal test set.

Any fault in an irredundant network causes growth of some X_i or Y_j term due to one or more literals. Procedure 2.1 first generates one variable less terms for each X_i and

Y_j term. The corresponding Z_k and W_m terms represent tests to detect all faults which cause some X_i or Y_j grow due to one variable, and most (if not all) of the faults which cause growth due to more than one literal. Step 3 guarantees that a test is also generated for any fault of the latter type for which a test may not have been generated in Step 2.

Minimality of T_s follows from the fact that MC_1 is a minimal covering of all W_k and Z_m terms. Because T_s is selected from MC_1 as a minimal test set detecting all s-0 and s-1 faults on the checkpoints it is also (Lemma 2.1) a minimal test set for the entire network.

The Equivalent Normal Form as used by Armstrong [2] is similar to the ESP form employed by procedure 2.1; however, there are some significant differences between the two methods. In [2], tests are generated sequentially and a scoring function is used to select the next test. The resulting test set is nearly minimal. Armstrong has conjectured, but not proved, that a complete test set will be generated. On the other hand, procedure 2.1 derives a minimal test set, and step 3 of this procedure guarantees the completeness of the test set.

<u>Example 2.1</u>: Consider the circuit in fig. 2.1 whose Karnaugh map is given in Table 2.1. Table 2.2 is obtained by performing steps 1 and 2 of procedure 2.1. Minimal covering MC_1, of all the Z_k and W_m terms can be obtained by inspection: $MC_1 = (abcd, a\bar{b}\bar{c}d, a\bar{b}c\bar{d}, \bar{a}b\bar{c}d, \bar{a}bc\bar{d}, abc\bar{d}, ab\bar{c}d, a\bar{b}cd, \bar{a}bcd)$, and step 5 yields Table 2.3. From this table, $MC_2 = (\bar{a}b\bar{c}d, abcd, abc\bar{d}, a\bar{b}cd) = T_s$ is a minimal single fault test set.

Procedure 2.1 derives a minimal test set that detects all single faults. This test set may not detect all multiple faults. The problem of multiple fault detection will be examined in the following chapter.

TABLE 2.1

Karnaugh Map for Network N2.1

cd \ ab	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	1	1	0	1
10	0	0	1	0

TABLE 2.3

Checkpoint Covering Table for Network N2.1

	1	2	3	4	5	6	7	8	
$abcd$	0	0	0	0	-	0	0	-	X
$a\bar{b}\bar{c}d$	-	1	1	-	1	-	-	1	
$a\bar{b}c\bar{d}$	-	1	-	1	1	-	-	1	
$\bar{a}b\bar{c}d$	1	-	1	-	1	-	-	1	X
$\bar{a}bc\bar{d}$	1	-	-	1	1	-	-	1	
$abc\bar{d}$	0	0	-	1	0	-	1	-	X
$ab\bar{c}d$	0	0	1	-	0	-	1	-	
$a\bar{b}cd$	-	1	0	0	-	1	-	0	X
$\bar{a}bcd$	1	-	0	0	-	1	-	0	

TABLE 2.2

Testing Table for Network N2.1

X_i	X_i'	$Z_k=X_i'.\bar{f}$	Covered by T_s
$a_{15}b_{25}\bar{c}_{37}$	ab	$abcd$	X
	$a\bar{c}$	$a\bar{b}\bar{c}$	
	$b\bar{c}$	$\bar{a}b\bar{c}$	X
$a_{15}b_{25}\bar{d}_{47}$	ab	$abcd$	X
	$a\bar{d}$	$a\bar{b}\bar{d}$	
	$b\bar{d}$	$\bar{a}b\bar{d}$	
$\bar{a}_{16}c_{38}d_{48}$	$\bar{a}c$	$\bar{a}c\bar{d}$	
	$\bar{a}d$	$\bar{a}\bar{c}d$	X
	cd	$abcd$	X
$\bar{b}_{26}c_{38}d_{48}$	$\bar{b}c$	$\bar{b}c\bar{d}$	
	$\bar{b}d$	$\bar{b}\bar{c}d$	
	cd	$abcd$	X

(Continued on p. 32)

TABLE 2.2 (Continued)

Y_j	Y_j'	$W_m=Y_j'.f$	Covered by T_s
$\bar{a}_{15}a_{16}b_{26}$	$\bar{a}b$ ab	$\bar{a}bcd$ $ab\bar{c},ab\bar{d}$	 X
$\bar{a}_{15}\bar{c}_{38}$	$\bar{a}$ $\bar{c}$	$\bar{a}cd$ $ab\bar{c}$	
$\bar{a}_{15}\bar{d}_{48}$	$\bar{a}$ $\bar{d}$	$\bar{a}cd$ $ab\bar{d}$	 X
$a_{16}b_{26}\bar{b}_{25}$	ab $a\bar{b}$	$ab\bar{c},ab\bar{d}$ $a\bar{b}cd$	X X
$\bar{b}_{25}\bar{d}_{48}$	$\bar{b}$ $\bar{d}$	$\bar{b}cd$ $ab\bar{d}$	X X
$\bar{b}_{25}\bar{c}_{38}$	$\bar{b}$ $\bar{c}$	$\bar{b}cd$ $ab\bar{c}$	X
$a_{16}b_{26}c_{37}d_{47}$	abc abd acd bcd	$abc\bar{d}$ $ab\bar{c}d$ $a\bar{b}cd$ $\bar{a}bcd$	X X
$\bar{c}_{38}c_{37}d_{47}$	cd $\bar{c}d$	$\bar{a}cd,\bar{b}cd$ $ab\bar{c}d$	X
$c_{37}d_{47}\bar{d}_{48}$	cd $c\bar{d}$	$\bar{a}cd,\bar{b}cd$ $abc\bar{d}$	X X

Chapter 3

MULTIPLE FAULT DETECTION IN IRREDUNDANT NETWORKS

The problem of how to detect all faults, single as well as multiple, has received much less attention than the single fault problem. One reason is that the total number of faults in a combinational network with n lines is 3^n-1 [33] and the complexity of the calculations necessary to consider all of them is tremendous. It will be shown, however, that only a very small fraction of this total number must be considered. Some faults (such as s-0 faults on the input and output lines of an AND gate) are indistinguishable. They can be combined into equivalence classes [43] and any test which detects a fault belonging to a class detects all faults in that class. From now on any equivalence class of single faults will be considered as a single fault. Even more important is the fact, as in the case of single faults, that only faults on the checkpoints must be considered. It will be proved that any multiple fault within a given network is equivalent to some multiple fault on the checkpoints of the same network. Two faults, F_1 and F_2, are equivalent if and only if the two Karnaugh maps representing the output function of the network in the presence of F_1 and F_2, respectively, are the same. The <u>multiplicity</u> of a multiple fault is defined to be the number of single faults which occur together. For example, the multiplicity of a single fault is 1, multiplicity of a multiple fault consisting of two single faults, is 2, etc.

<u>Lemma 3.1</u>: [5] Any multiple fault in a combinational network with n checkpoints is equivalent to some multiple fault of multiplicity k among the checkpoints only, where $1 \leq k \leq n$.

<u>Proof</u>: (i) Networks without internal fanout have a tree-like structure, where the gates represent the nodes and the primary output is the root of the tree. A s-0 (s-1) fault on the output line of a gate is equivalent to some s-0 or s-1 fault(s) on one or more inputs of that gate. Consequently, it is possible to find an equivalent fault pattern on the input checkpoints of the network. The multiplicity of this new pattern cannot be greater than the total number of checkpoints, which is n. The two fault patterns are equivalent in the sense that their effect upon the network function is the same and any test detecting the input fault pattern detects also the original pattern. Hence, if all multiple faults among the input checkpoints are detected, all multiple faults within the network are detected as well.

(ii) Networks with internal fanout can be decomposed into fanout-free segments. The proof follows from

(i) above, since all the fanout branches are checkpoints.

Theorem 3.1: In order to detect all multiple faults in an irredundant combinational network it is necessary and sufficient to detect all multiple faults on its checkpoints.

Proof: The proof follows from the previous lemma.

3.1 Fault Masking and Minimal Test Sets for Multiple Faults

There is no need to consider all multiple faults on the checkpoints of the network. A test set for single faults of a given network does not detect those multiple faults that consist of faults which prevent detection of each other. This effect is called fault masking (see [17], [18]). Before going into more detail, some notation will be introduced:

p-1 denotes an s-1 fault on checkpoint p,

p-0 denotes an s-0 fault on checkpoint p,

f_k denotes a single fault,

F_k denotes a multiple fault,

(f_1,f_2) denotes multiple fault consisting of single faults f_1 and f_2.

$f_1:X_i \to X_i', Y_j \to Y_j'$ denotes a fault f_1 causing growth of X_i to X_i' and Y_j to Y_j',

$(f_1,f_2):Y_j \to 0$ denotes the multiple fault (f_1,f_2), causing the term Y_j to disappear.

<u>Definition 3.1</u>: A single or multiple fault F_1 is a <u>subfault</u> of a multiple fault $F_2 = (f_1, f_2, \ldots, f_n)$ if F_1 is a subset of F_2.

In a fanout-free network, the multiple fault $F = (f_1, f_2, \ldots, f_n)$ causes the disappearance of the term X_i, if at least one fault f_k, $k = 1,2,\ldots, n$, makes X_i disappear. Conversely, F makes X_i grow if no fault f_k, $k \varepsilon 1,2,\ldots, n$, causes X_i to disappear and at least one fault f_m, $m \varepsilon 1,2,\ldots, n$, makes X_i independent of some of its variables. In the case of networks with internal fanout, the effect of a multiple fault upon some X_i or Y_j term must be established more carefully. For example, for the network in fig. 2.1 the term $X_1 = a_{15}b_{25}\bar{c}_{37}$ disappears due to fault (1-0,2-0), but it grows into $X_1' = \bar{c}_{37}$ due to fault (1-0,2-0, 5-1). This term would also disappear due to fault (5-1,3-1) or fault (5-1,7-1).

<u>Definition 3.2</u>: A fault <u>f_2</u> is said to <u>mask</u> a fault <u>f_1</u> (denoted f_2/f_1) under a test set T, if any test t in T that detects f_1 when it occurs alone, does not detect the occurrance of f_1 and f_2 together, i.e., (f_1, f_2).

Let fault f_1 cause the growth of X_i into X_i', and let test t be contained in the intersection $X_i'.\bar{f}$, where $\bar{f}$ is the complement of the output function f. In other words t detects f_1, since $f(t) = 0$ in the fault-free network and $f(t) = 1$ in the presence of f_1. If some fault f_2 masks fault f_1, then the output of the network under test t in the presence of f_1 and f_2 must be the same as the output of the fault-free network, i.e. $f(t) = 0$ in the presence of (f_1, f_2). From this fact, some important relations between faults capable of masking and the X_i and Y_j terms can be deduced. These relations are stated in Theorem 3.2.

<u>Theorem 3.2</u>: Let fault f_1 be detected by test t_x (or test t_y) which checks for growth of $X_1, X_2, \ldots, X_p$ (or $Y_1, Y_2, \ldots, Y_r$) due to f_1. If fault f_2 masks fault f_1 under test t_x (or t_y), then:

(a) f_1 detected by t_x 1) $f_2: X_1 \rightarrow 0, \ldots, X_p \rightarrow 0$, and
2) there exists a term Y_α such that Y_α grows into Y'_α and $t_x \in Y'_\alpha$

(b) f_1 detected by t_y 1) $f_2: Y_1 \rightarrow 0, Y_2 \rightarrow 0, \ldots, Y_r \rightarrow 0$, and
2) there exists a term X_β such that X_β grows into X'_β and $t_y \in X'_\beta$.

<u>Proof</u>: Only case (a) will be proved, since the proof for (b) follows by duality.

1. Assume f_1 causes the growth of X_1 only. Hence if f_1 occurs $X_1 \rightarrow X'_1$, but $t_x \in X'_1 . \bar{f}$ and f_1 is detected, unless $X_1 = 0$ due to some other fault. If f_1 causes growth of $X_1, X_2, \ldots, X_p$ which are checked for growth due to fault f_1 by test t_x, then f_2 must cause disappearance of all of them in order to mask f_1.

2. If only f_1 occurs, the output of the network under t_x is 1, and consequently $f_1: Y_j \rightarrow 0$ for all Y_j such that $t_x \in Y_j$.

When (f_1,f_2) occurs then $f_2:X_1 \to 0,\ldots,X_p \to 0$ and some Y_α must grow to Y'_α and $t_x \varepsilon Y'_\alpha$ because the output of the network under input t_x is the same as the output of the fault-free network. Provided that f_1 and f_2 are the only faults that occur in the network, then Y_α grows due to f_2 or (f_1,f_2). If some other faults occur simultaneously with f_1 and f_2, then Y_α may also grow due to such faults.

A single fault can be masked by both single and multiple faults. To illustrate, suppose that fault f_1 is detected by a test t that checks for growth, due to f_1, of $X_1,X_2\ldots,X_p$. It is possible that no single fault causes the disappearance of all terms $X_1,X_2,\ldots,X_p$; however, if some multiple fault F_2 occurs such that $F_2:X_1 \to 0,\ldots, X_p \to 0$, then f_1 will not be detected when f_1 and F_2 both occur. It should be pointed out that, when masking is investigated, the necessary and sufficient conditions for masking to occur are the important features. Obviously, if some subfault of F_2 masks fault f_1 as well, then the occurrence of F_2 is sufficient but not necessary for f_1 to be masked. From now on, by saying that a multiple fault F_2 masks a fault f_1, it is understood that F_2 masks f_1, but no subfault of F_2 masks f_1. Corrollary 3.1 restates the result of theorem 3.2 for this more general case (only one of the two dual versions is stated).

Corollary 3.1: Let fault f_1 be detected by a test t that checks for growth, due to f_1, of $X_1,X_2,\ldots,X_p$. If a multiple fault F_2 masks fault f_1 under test t, then

1. F_2:$X_1 \rightarrow 0,\ldots, X_p \rightarrow 0$, and
2. There exists a term Y_α such that Y_α grows into Y'_α and $t \in Y'_\alpha$.

If a fault F_a masks n individual faults, say faults $f_1,f_2,\ldots,f_n$, then F_a causes the disappearance of all X_i or Y_j terms that grow due to these faults. Consequently, F_a will also mask any fault F that is a subset of $\{f_1,f_2,\ldots,f_n\}$.

Multiple faults that are not detected by a single fault test set will now be considered. Let T_s be a single fault test set for the given network. A multiple fault $F = (f_1,f_2,\ldots,f_n)$ is not detected by T_s if every fault f_k, $k = 1,2,\ldots,n$ is masked, under T_s, by some subfault of F. Suppose that fault f_k is detected by a test t_k such that the output function $f(t_k) = 0$ in the fault-free network. If fault f_k is masked, then, by theorem 3.2 there exists some Y_j term that grows into Y'_j and $t_k \in Y'_j$; however, if the intersection $Y'_j.f$ contains at least one test $t_i \in T_s$, then the growth of this term will be detected at the output. Consequently, when looking for multiple faults that are not detected by T_s, it is sufficient to consider only those X_i or Y_j terms that are not checked by T_s for growth due to some fault.

It is convenient to use an oriented graph for describing the masking relations among a set of faults under a given test set. The nodes of the graph are the faults considered and an arc from node A to node B indicates that fault A masks fault B under the given test set. Assuming that the single fault test set T_s has been already derived, the following procedure constructs the masking graph.

<u>Procedure 3.1</u>:

1. For any X_i or Y_j term that is not checked for growth due to some fault perform step 2.

2. For any single or multiple fault F such that $F: X_i \rightarrow X_i'$ and $X_i'.\bar{f}.T_s = \emptyset$ (or $F: Y_j \rightarrow Y_j'$ and $Y_j'.f.T_s = \emptyset$) construct the intersection

$$P = X_i'.\bar{X}_i.f.T_s \quad (\text{or } Y_j'.\bar{Y}_j.f.T_s)$$

If $P = \{t_i\} \neq \emptyset$, consider the set S of all faults detected by $\{t_i\}$. Remove from S all faults that cause term X_i' (or Y_j') to disappear. Then use the first condition of theorem 3.2, or corrollary 3.1, to determine the masking relations, under $\{t_i\}$, between any fault $f_a \in S$ and any subfault of F. Disregard any masking relations between two faults on inputs of the same gate.

3. When step 2 has been performed for all X_i and Y_j terms, construct the masking graph from the masking relations obtained above.

The validity of the procedure is established in Appendix A. An example follows, that demonstrates the use of this procedure.

Example 3.1: For the network in fig. 3.1 test set T_{s1},

$T_{s1} = (\bar{a}\bar{b}c\bar{d}egh, abcde\bar{g}\bar{h}, \bar{a}bcd\bar{e}\bar{g}h, a\bar{b}cd\bar{e}\bar{g}h, ab\bar{c}ed\bar{g}h,$

$ab\bar{c}\bar{d}eg\bar{h}, ab\bar{c}d\bar{e}g\bar{h}) = (t_1,t_2,t_3,t_4,t_5,t_6,t_7)$

detects all single faults (see table 3.1). It is to be determined whether T_{s1} detects also all multiple faults. Table 3.2 lists all the X_i and Y_j terms of the two ESP forms, the terms X_i' and Y_j', and the corresponding Z_k and W_m terms, where $Z_k = X_i'.\bar{f}$ and $W_m = Y_j'.f$. The table also shows how these Z_k and W_m terms are covered by test set T_{s1}. The term $X_1 = a_1b_2c_3d_4e_5$ is not checked for growth due to faults 1-1, 2-1, 4-1 and 5-1. X_1 grows due to the multiple fault (1-1,2-1,4-1,5-1) into $X_1' = c_3$, but the intersection $X_1'.\bar{f}.T_{s1}$ contains test t_3 and t_4 and consequently this growth will be checked at the output. Next, it is necessary to check how X_1 grows due to all faults that are subsets of (1-1,2-1,4-1,5-1). Due to fault (1-1,2-1,4-1) X_1 grows into $X_1' = c_3e_5$, where $X_1'.\bar{f}.T_{s1} = \emptyset$ and the intersection P is

$$P = X_1'.\bar{X}_1.f.T_{s1} = ce.(\bar{a}+\bar{b}+\bar{d}).f.T_{s1} = t_1$$

Test t_1 detects faults 6-0 and 7-0. Neither fault causes X_1 to disappear. Test t_1 detects fault 6-0 by checking for the growth of $Y_1 = \bar{a}_1\bar{g}_6$ and $Y_2 = \bar{b}_2\bar{g}_6$ (see table 3.2) and both

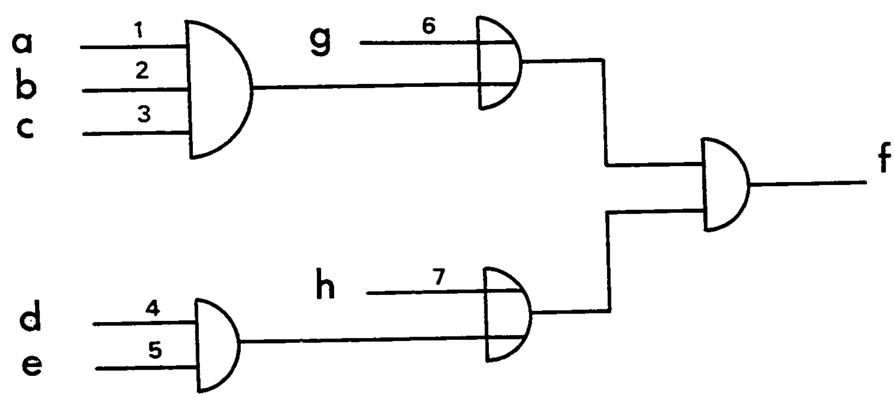


Figure 3.1 Network N3.1

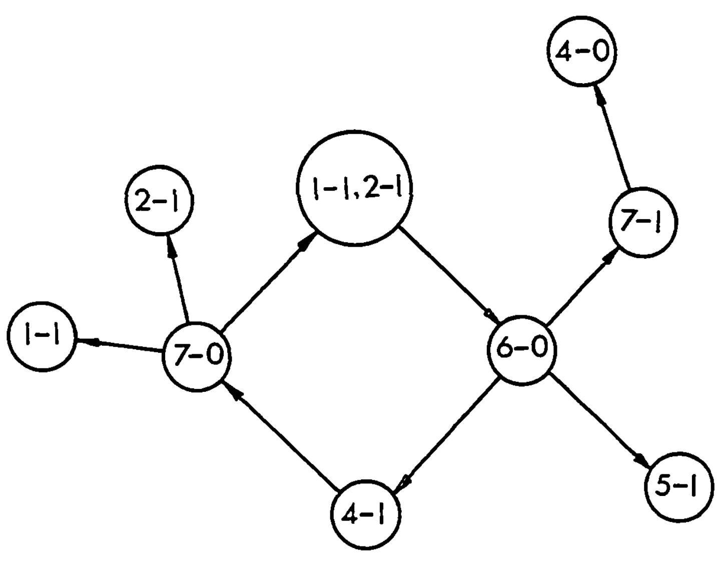


Figure 3.2 Masking graph for network N3.1

TABLE 3.1

Test Set T_{s1} for Network N3.1

	1	2	3	4	5	6	7
$t_1=\bar{a}\bar{b}c\bar{d}egh$	-	-	-	-	-	0	0
$t_2=abcde\bar{g}\bar{h}$	0	0	0	0	0	-	-
$t_3=\bar{a}bcd\bar{e}\bar{g}h$	1	-	-	-	-	1	-
$t_4=a\bar{b}cd\bar{e}\bar{g}h$	-	1	-	-	-	1	-
$t_5=ab\bar{c}de\bar{g}h$	-	-	1	-	-	1	-
$t_6=ab\bar{c}\bar{d}eg\bar{h}$	-	-	-	1	-	-	1
$t_7=ab\bar{c}d\bar{e}g\bar{h}$	-	-	-	-	1	-	1

TABLE 3.2

Testing Table for Network N3.1

X_i	X_i'	$Z_k=X_i'.\bar{f}$	Covered by T_{s1}
$a_1b_2c_3d_4e_5$	abcd	$abcd\bar{e}\bar{h}$	
	abce	$abc\bar{d}e\bar{h}$	
	abde	$ab\bar{c}de\bar{g}$	X
	acde	$a\bar{b}cde\bar{g}$	
	bcde	$\bar{a}bcde\bar{g}$	
$a_1b_2c_3h_7$	abc	$abc(\bar{d}\bar{h}+\bar{e}\bar{h})$	
	abh	$ab\bar{c}\bar{g}h$	X
	ach	$a\bar{b}c\bar{g}h$	X
	bch	$\bar{a}bc\bar{g}h$	X
$d_4e_5g_6$	de	$(\bar{a}+\bar{b}+\bar{c})de\bar{g}$	X
	dg	$d\bar{e}g\bar{h}$	X
	eg	$\bar{d}eg\bar{h}$	X
g_6h_7	g	$(\bar{d}+\bar{e})g\bar{h}$	X
	h	$(\bar{a}+\bar{b}+\bar{c})\bar{g}h$	X

Y_j	Y_j'	$W_m=Y_j'.f$	Covered by T_{s1}
$\bar{a}_1\bar{g}_6$	$\bar{a}$	$\bar{a}g(de+h)$	X
	$\bar{g}$	$abc\bar{g}(de+h)$	X
$\bar{b}_2\bar{g}_6$	$\bar{b}$	$\bar{b}g(de+h)$	X
	$\bar{g}$	$abc\bar{g}(de+h)$	X
$\bar{c}_3\bar{g}_6$	$\bar{c}$	$\bar{c}g(de+h)$	
	$\bar{g}$	$abc\bar{g}(de+h)$	X
$\bar{d}_4\bar{h}_7$	$\bar{d}$	$(abc+g)\bar{d}h$	X
	$\bar{h}$	$(abc+g)de\bar{h}$	X
$e_5\bar{h}_7$	$\bar{e}$	$(abc+g)\bar{e}h$	
	$\bar{h}$	$(abc+g)de\bar{h}$	X

terms disappear due to the multiple fault (1-1,2-1). But, test t_1 is the only test that detects fault 6-0 and consequently (1-1,2-1)/6-0 under T_{s1}. Fault 7-0 is detected by checking for the growth of $Y_4 = \bar{d}_4\bar{h}_7$. This term disappears due to fault 4-1, hence 4-1/7-0. Because test t_1 is the only test that is contained in the intersection $\bar{X}_1.f.T_{s1}$, there is no need to consider any other term X_1'. By repeating step 2 of procedure 3.1 with terms X_2,Y_3 and Y_5, the following masking relations are obtained: 7-1/4-0 (here fault 4-0 is chosen to represent the equivalence class (4-0,5-0)), 6-0/4-1, 6-0/5-1, 6-0/7-1, 7-0/1-1 and 7-0/2-1. In addition, fault 7-0 masks fault 6-1 under t_3 and t_4. This relation is disregarded, because 7-0 does not mask 6-1 under t_5 and consequently under T_{s1}. The masking graph obtained is given in fig. 3.2. From this graph it is easy to observe that the multiple fault F_1 = (1-1,2-1,4-1,6-0,7-0) is not detected by T_{s1}. The graph also indicates that all subfaults of the multiple fault F_2 = (1-1,2-1,4-1,6-0,7-0,7-1) are masked. This fault however, can be disregarded, because checkpoint 7 cannot be both s-0 and s-1 at the same time. The same is true for multiple fault F_3 = (1-1,2-1,4-1,5-1,6-0,7-0), since fault (4-1,5-1) is equivalent to fault 7-1. The multiple fault F_1 can be detected by any test that checks for growth of some X_i or Y_j term that grows due to F_1. For example, test $\bar{a}\bar{b}\bar{c}\bar{d}\bar{e}gh$ detects this fault, because F_1 causes growth of $Y_3 = \bar{c}_3\bar{g}_6$ into $Y_3' = \bar{c}_3$ and $\bar{a}\bar{b}\bar{c}\bar{d}\bar{e}gh$ is contained in the intersection $Y_3'.f$. It should be noted that test set T_{s1} was selected from a highly nonminimal covering of the Z_k and W_m terms. Procedure SF would produce test set T_{s2},

$$T_{s2} = (\bar{a}bcde\bar{g}h,\ \bar{a}bcde\bar{g}h,\ ab\bar{c}de\bar{g}h,\ abc\bar{d}eg\bar{h},\ abcd\bar{e}g\bar{h},\ abcde\bar{g}\bar{h},\ \bar{a}\bar{b}\bar{c}\bar{d}\bar{e}gh)$$

and this test set detects all single as well as all multiple faults.

If a multiple fault exists, that is not detected by the single fault test set, then the masking graph has a strongly connected subgraph, i.e. there is at least one loop in the graph. If the masking graph does not have any loops, then the test set detects all multiple faults, since at least one subfault of any multiple fault is not masked. For the above example, note that the multiple fault (6-0,7-1) is detected, because 6-0 is not masked if (1-1,2-1) does not occur, etc.

After finding the multiple faults that are not detected by the single fault test set T_s, this test set is enlarged to detect the undetected multiple faults. Procedure 3.2 given below derives a minimal multiple fault test set T_m by adding a minimal number of additional tests to the test set T_s generated by procedure 2.1.

Procedure 3.2:

1. Perform steps 1-6 of procedure 2.1 to derive the minimal single fault test set T_s.

2. Use procedure 3.1 to construct the masking graph and find the set $M = (F_1, F_2, \ldots F_R)$ of multiple faults that are not detected by T_s. If M contains only one fault, find a test t that detects the fault, set T' = t and go to 5.

3. For each fault F_p in M and for all X_i and Y_j terms that grow due to F_p compute

X_i' such that $F_p: X_i \rightarrow X_i'$, and $Z_k = X_i' . \bar{f}$

Y_j' such that $F_p: Y_j \rightarrow Y_j'$, and $W_m = Y_j' . f$

and the set D_p of all tests that detect fault F_p is D_p = {union of all Z_k and W_m terms computed above}.

4. Find a minimal test set T' such that it covers all the D_p terms.

5. The multiple fault test set T_m is given by

$T_m = T_s \cup T'$.

The multiple fault set test derived by this procedure is a minimal test set in that it contains the single fault test set T_s as a subset. For a given network, there may be a number of minimal single fault test sets, $T_{s1}, T_{s2}, \ldots, T_{sn}$ (all of the same length). The associated multiple fault test sets $T_{m1} \supset T_{s1}, \ldots, T_{mn} \supset T_{sn}$, however, do not have to be of the same length. If a single fault test set T_{si} checks for growth of as many X_i and Y_j terms as possible, then the number of faults capable of masking and also the number of undetected multiple faults is minimized. Consequently, the associated multiple fault test T_{mi} will be minimal. It is felt, that a necessary and sufficient condition for choosing T_{si} with the above properties is as specified in step 4 of the procedure 2.1, i.e., that MC_1 is a minimal cover (rather than any cover) of all the Z_k and W_m terms.

Example 3.2: For the network in fig. 2.1 (see p.13), test set T_s derived by procedure 2.1 is

$$T_s = (\bar{a}b\bar{c}d,\ abcd,\ a\bar{b}cd,\ abc\bar{d}).$$

Table 2.2 (see p.31) will be used when performing procedure 3.1. Step 2 of this procedure applied to the four X_i terms reveals no masking, since the intersection P is always empty. Term $Y_1 = \bar{a}_{15}a_{16}b_{26}$ is not checked for growth due to fault 6-1 (see Table 2.2) and it can grow into $Y_1' = \bar{a}_{15}$. The intersection P contains test $\bar{a}b\bar{c}d$ and the intersection $Y_1'.f.T_s$ is empty. Test $\bar{a}b\bar{c}d$ detects faults 1-1,3-1,5-1 and 8-1 (see Table 2.3). Faults 1-1 and 5-1 cause Y_1' to disappear, and fault 6-1 belongs to the same gate as 8-1. Hence the only relation that is marked in the masking graph (see fig. 3.3) is 6-1 masks 3-1. Term $Y_2 = \bar{a}_{15}\bar{c}_{38}$ can grow into $\bar{a}_{15}$ and $\bar{c}_{38}$, respectively, but the intersection P is empty in both cases. Step 2 performed with term

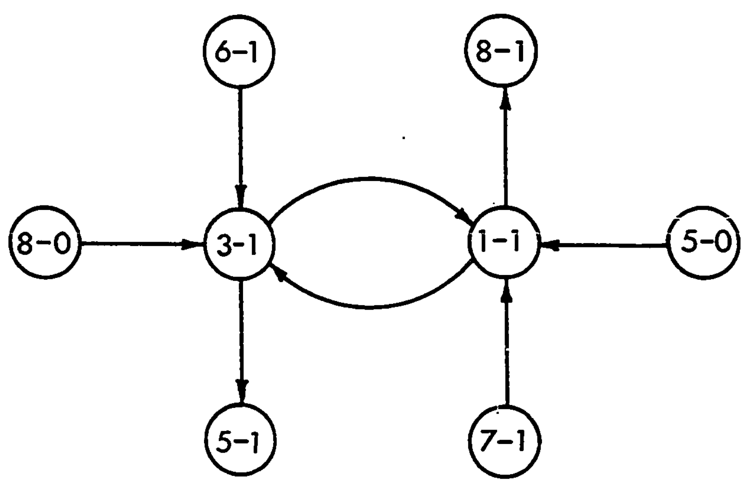


Figure 3.3 Masking graph for network N2.1

$Y_3' = \bar{a}_{15}$ shows that fault 8-0 masks fault 3-1, and the same argument with term $Y_6' = \bar{c}_{38}$ reveals that 5-0 masks 1-1. Term $Y_7 = a_{16}b_{26}c_{37}d_{47}$ is not checked for growth due to fault 1-1 and fault 3-1. For the term $Y_7' = b_{26}d_{47}$, $P = \bar{a}b\bar{c}d$ and $Y_7'.f.T_s = \emptyset$. Test $\bar{a}b\bar{c}d$ detects fault 1-1 and fault 5-1 by checking for growth of $X_1 = a_{15}b_{25}\bar{c}_3$. Since 3-1 causes disappearance of X_1, 3-1 masks fault 1-1 and fault 5-1. The same reasoning with term $X_3 = \bar{a}_{16}c_{38}d_{48}$ shows that fault 1-1 masks fault 3-1 and fault 8-1. Finally, step 2 of procedure 3.1 performed with term $Y_8' = \bar{c}_{38}$ indicates that 7-1 masks 1-1. The masking graph constructed from the above masking relations is in fig. 3.3. From this graph, it is seen that the following multiple faults are not detected: (1-1,3-1), (1-1,3-1,5-1), (1-1,3-1,8-1), (1-1,3-1,5-1,8-1). Any one of the faults causes growth of Y_7 into $Y_7' = b_{26}d_{47}$ and $Y_7'.f = \bar{a}bcd + ab\bar{c}d$.

Test set T_m

$$T_m = (\bar{a}b\bar{c}d,\ abcd,\ a\bar{b}cd,\ abc\bar{d},\ \bar{a}bcd)$$

is one of the several minimal test sets that detect all single and multiple faults.

<u>Example 3.3</u>: For the network in fig. 3.4, the two ESP forms are:

$$ESP(f) = \bar{a}_{17}b_4 + b_4\bar{c}_{27}d_{37} + a_{18}\bar{b}_5c_{28} + a_{18}\bar{b}_5\bar{d}_{38} + a_{18}c_{28}\bar{d}_6 + a_{18}\bar{d}_{38}\bar{d}_6$$

$$ESP(\bar{f}) = \bar{b}_4b_5d_6 + \bar{a}_{18}\bar{b}_4 + \bar{b}_4\bar{c}_{28}d_{38} + a_{17}b_5c_{27}d_6 + a_{17}\bar{a}_{18}c_{27} +$$

$$+\ a_{17}c_{27}\bar{c}_{28}d_{38} + a_{17}b_5d_6\bar{d}_{37} + \bar{a}_{18}a_{17}\bar{d}_{37} + a_{17}\bar{c}_{28}d_{38}\bar{d}_{37}$$

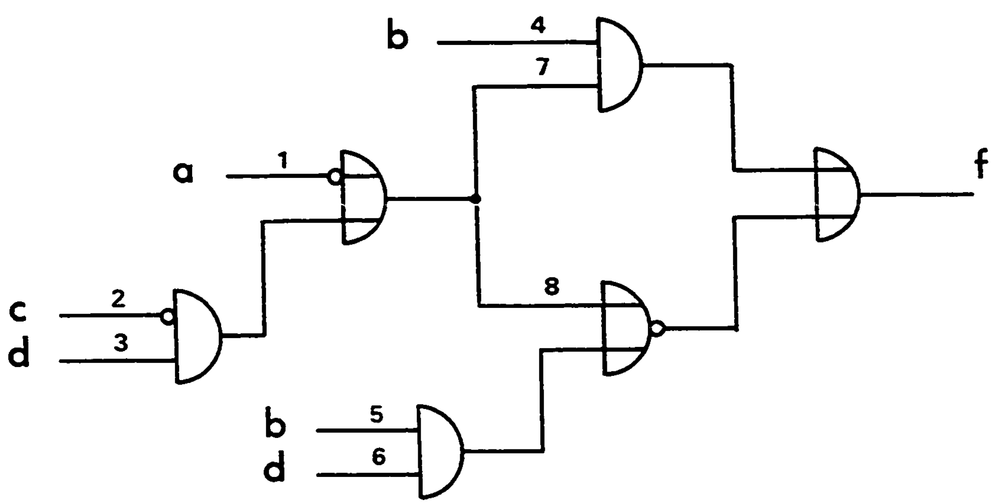


Figure 3.4 Network N3.4

By procedure 2.1, MC_1 = ($abcd$, $a\bar{b}\bar{c}d$, $\bar{a}\bar{b}cd$, $a\bar{b}\bar{c}\bar{d}$, $a\bar{b}cd$, $ab\bar{c}d$, $abc\bar{d}$, $\bar{a}bcd$, $\bar{a}bc\bar{d}$) and T_s = ($a\bar{b}\bar{c}\bar{d}$, $a\bar{b}cd$, $abcd$, $abc\bar{d}$, $a\bar{b}\bar{c}d$, $\bar{a}bcd$). Procedure 3.1 shows that there are no undetected multiple faults and $T_m = T_s$ is a minimal multiple fault test set for the network.

Procedure NR in [5] is claimed to generate a nearly minimal multiple fault test set for any irredundant network. The network in fig. 3.4 is a counterexample . By procedure NR Abnormal True Tests = ($abcd$, $a\bar{b}\bar{c}d$, $\bar{a}\bar{b}cd$)
Tests ($ab\bar{c}\bar{d}$, $a\bar{b}cd$, $ab\bar{c}d$) verify that checkpoints are "normal." This test set does not detect all faults, because test $a\bar{b}\bar{c}\bar{d}$, which is the only test detecting the fault 3-1, is not contained in the set.

The method presented in this section generates a multiple fault test set which is minimal. However, the cost of obtaining this minimal set is reflected in the increased computational complexity. In the following section a considerably simpler method to derive a nearly minimal test set will be described.

3.2 Sufficient Conditions for Multiple Fault Detection

Rather than looking for multiple faults that are not detected by a given single fault test set, it is possible to use a sufficient although not a necessary condition (theorem 3.3) and derive the multiple fault test set directly. We shall first consider networks with an ESP form such that all terms can be checked for growth as defined in the following definition.

Definition 3.3: Growth of a term X_i is strongly checked by a test set T, if for any single or multiple fault F_k, such that $F_k:X_i \to X_i'$, the intersection $X_i'.\bar{f}$ is nonempty and contains at least one test t ε T.

A similar definition for growth of a Y_j term follows by duality. The following theorem specifies sufficient conditions for multiple fault detection in irredundant networks.

Theorem 3.3: Let N be an irredundant network with $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$. If a test set T detects all single faults in N and strongly checks for growth of either all X_i terms, or all Y_j terms, then T detects all multiple faults in N.

Proof: Let test set T detect all single faults in N. In order to prove that T detects all multiple faults it is sufficient to show that the masking graph, under T, does not have any cycles. Suppose that test set T checks strongly for growth of all X_i terms, and let T_a and T_b define a partition on T such that all tests that produce the output of one in the fault-free network are in T_a.

Any fault f_a that is detected by some test $t_a \varepsilon T_a$ causes growth of some Y_j into Y_j', where $t_a \varepsilon Y_j'.f$. By theorem 3.2, if f_a is masked there exists term X_i that grows into X_i' and $t_a \varepsilon X_i'$. However, since T strongly checks for growth of all X_i terms, the intersection $X_i'.\bar{f}$ will contain at least one test from T_b and consequently growth of any X_i term will be reported at the output. Hence, if any fault f_a that is detected by some test from T_a is masked, then it is masked by a fault that cannot be masked under test set T. In other words, no fault f_a can belong to a cycle of the masking graph.

Let fault f_b cause growth of $X_1, X_2, \ldots, X_p$. Fault f_b is masked under test set T, if there occurs a fault f_2 (or F_2) that makes all the terms $X_1, X_2, \ldots, X_p$ disappear. It is sufficient to consider that f_b is masked by f_2; if f_b is masked by F_2, then the following argument can be applied to all components of F_2. Two possibilities can occur:

i) Fault f_2 is detected by some test from T_a. Then fault f_2 cannot belong to a cycle of the masking graph as was shown above.

ii) Fault f_2 is detected by some test from T_b, and then it causes the growth of one or more X_i terms, say terms $X_{p+1}, \ldots, X_{p+r}$. Fault f_2 is detected by T, unless there exists fault f_3 (or F_3) such that it causes $X_{p+1}, \ldots, X_{p+r}$ to disappear. Argument (i) and (ii) can now be repeated for fault f_3. Obviously, any multiple fault either contains at least one fault that is not masked under T, or makes all the X_i terms disappear and then the output of the network would be constant.

For any network with at least one ESP form such that all its terms can be strongly checked for growth, the result of theorem 3.3 can be used to derive a nearly minimal multiple fault test set which will be minimal sometimes. The set of networks with an irredundant ESP form is a proper subset of all such networks. Because an irredundant ESP form does not contain any redundant terms or literals, it is possible to check strongly for growth of all its terms. The multiple fault test set is generated in the following two steps.

Procedure 3.3:

1. Use procedure 2.1 to derive a minimal single fault test set T_s.

2. If T_s strongly checks for growth of all X_i or all Y_j terms, then T_s is also a minimal multiple fault test set. Otherwise enlarge T_s so that either growth of all X_i terms or growth of all Y_j terms is strongly checked. The resulting test set is a nearly minimal multiple fault test set.

The above procedure derives the multiple fault test set without performing the multiple fault analysis and for this reason the computational effort is low. This is illustrated by the following two examples.

Example 3.3: Test set T

$T = (a\bar{b}\bar{c}d,\ \bar{a}b\bar{c}\bar{d}e,\ abc\bar{d},\ a\bar{b}\bar{c}\bar{d}e,\ ab\bar{c}d,\ ab\bar{c}d\bar{e},\ \bar{a}\bar{b}c\bar{d}\bar{e},\ ab\bar{c}\bar{d}e)$

is a minimal single fault test set for the network in fig. 3.5. The ESP(f) form of this network is

$ESP(f) = a_1\bar{a}_3 + a_1\bar{b}_4 + \bar{a}_3b_2 + b_2\bar{b}_4 + b_6c_5 + c_5d_7 + \bar{c}_8\bar{d}_9\bar{e}_{10}$

Because T strongly checks for growth of all X_i terms, it is also a minimal multiple fault test set.

Example 3.4: Test set $T_s = (\bar{a}b\bar{c}d,\ abcd,\ a\bar{b}cd,\ abc\bar{d})$ detects all single faults in the network in fig. 2.1. If tests $\bar{a}bcd$ and $ab\bar{c}d$ are added to T_s, all the Y_j terms are strongly checked for growth. Test set T_m

$$T_m = (\bar{a}b\bar{c}d,\ abcd,\ a\bar{b}cd,\ abc\bar{d},\ \bar{a}bcd,\ ab\bar{c}d)$$

is a nearly minimal multiple fault test set, since the length of a minimal multiple fault test set is five.

If a network has a redundant ESP form, then it is not always possible to strongly check for growth of all X_i or all Y_j terms. It should be noted, that the network itself does not have to be redundant; however, it is always possible to weakly check as defined below.

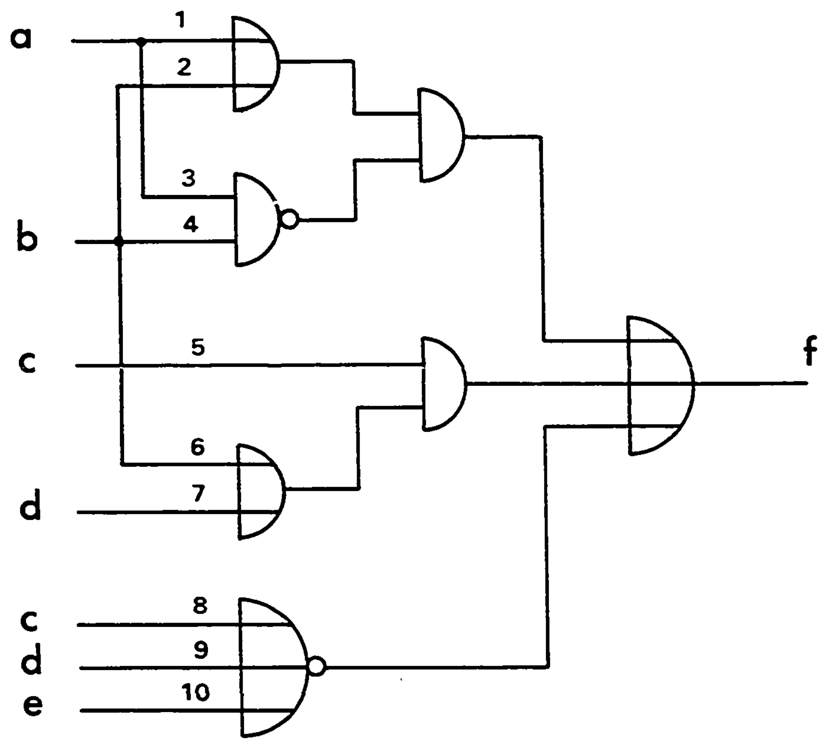


Figure 3.5 Network N3.5

<u>Definition 3.4</u>: Growth of a term X_i is <u>weakly checked</u> by test set T, if for any single or multiple fault F_k such that $F_k: X_i \rightarrow X_i'$, the intersection $X_i'.f$ is either empty, or if nonempty then it contains at least one test $t \in T$.

For example, for the network in fig. 3.4 the ESP(f) form is

$$ESP(f) = \bar{a}_{17}b_4+b_4\bar{c}_{27}d_{37}+a_{18}\bar{b}_5c_{28}+a_{18}\bar{b}_5\bar{d}_{38}+a_{18}c_{28}\bar{d}_6+a_{18}\bar{d}_{38}\bar{d}_6$$

Here, it is not possible to strongly check for growth of the term $X_4 = a_{18}\bar{b}_5\bar{d}_{38}$, since X_4 can grow, due to fault 5-0, into $X_4' = a_{18}\bar{d}_{38}$ and the intersection $X_4'.\bar{f}$ is empty. Tests $a\bar{b}\bar{c}d$ and $\bar{a}\bar{b}\bar{c}\bar{d}$ are sufficient to weakly check for growth of X_4.

It is difficult to prove, that a test set which detects all single faults and weakly checks for growth of all X_i terms (or all Y_j terms), also detects all multiple faults. However, a test set that weakly checks for growth of both the X_i terms and Y_j terms detects all faults in the network. Moreover, this result holds for irredundant as well as redundant networks.

<u>Theorem 3.4</u>: Let N be a combinational network with $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$. If test set T weakly checks for growth of all X_i and all Y_j terms, then T detects all detectable faults in N.

Proof: Any single or multiple fault F in a combinational network causes growth of one or more X_i and/or Y_j term. Provided that F is detectable, there must exist at least one X_i (or Y_j) term, such that F: $X_i \rightarrow X_i'$ and $X_i'.\bar{f} \neq \emptyset$. From definition 3.4, any such nonempty intersection must contain at least one test $t \in T$. Because test set T weakly checks for growth of all X_i and Y_j terms, it must detect all detectable faults. Fault F (single or multiple fault) is undetectable if F: $X_1 \rightarrow X_1', \ldots, X_n \rightarrow X_n', Y_1 \rightarrow Y_1', \ldots, Y_m \rightarrow Y_m'$ and all the intersections $X_1'.\bar{f}, \ldots, X_n'.\bar{f}, Y_1'.f, \ldots, Y_m'.f$ are empty.

Bossen and Hong [5] obtained a result equivalent to that one of theorem 3.4 by performing "cause-effect analysis for multiple fault detection", and they described a simple method (procedure G in [5]) that derives a multiple fault test set for any combinational network. However the test set generated by this procedure may be far from minimal.

Chapter 4

FAULT DETECTION IN REDUNDANT NETWORKS

In a redundant network certain faults are not detectable, hence the output of the network is correct in the presence of such faults. To ensure that the output function realized by a redundant network is correct, it is necessary to detect all detectable faults. It may also be desirable to detect all faults, not only the detectable ones, since the presence of an undetectable fault removes some of the redundant properties of the network. For this purpose, the network would have to be supplied with additional testing points, being in effect converted to a multiple output irredundant network. This problem will not be considered herein.

A redundant network contains at least one undetectable fault. Because any stuck fault on the output of a gate is equivalent to a multiple fault on the inputs of the same gate, a redundant network must contain at least one undetectable fault on some of its checkpoints. The set of undetectable faults on the checkpoints characterizes the redundant properties of the network in the sense, that corresponding to any single undetectable fault in the network there is an equivalent multiple fault consisting of undetectable faults on the checkpoints.

4.1 Single Fault Detection

When deriving a single fault test set for a redundant network, it is not sufficient to generate tests that detect all detectable single faults on the checkpoints. The test set derived in this way might not detect all detectable single faults in the network.

First consider fanout-free networks. For example, the network in fig. 4.1 has five checkpoints enumerated 1-5. The remaining lines in the network are denoted by the numbers 6-9. All single faults 1-0,2-0, 3-0,4-0,6-0 and 7-0 are undetectable. The fault 8-0, however, is detectable and any complete test set must contain a test for this fault.

Throughout this chapter denote

D - set of all detectable single faults on the checkpoints

U - set of all undetectable single faults on the checkpoints.

Lemma 4.1: Let N be a fanout-free redundant network. If a test set T detects all single faults in D and all detectable multiple faults that consist of single faults from U, then T detects all single detectable faults in N.

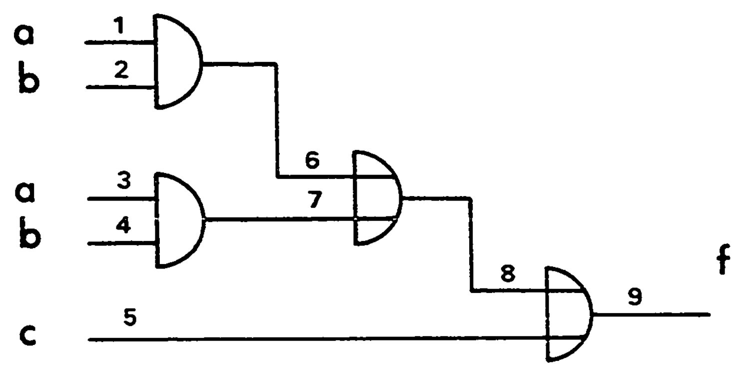


Figure 4.1 Network N4.1

<u>Proof</u>: Let T' be the subset of T that detects all single faults in D. If some detectable fault f_a exists that is not detected by T', then f_a can occur along only those paths that connect some undetectable faults on the checkpoints, say faults $f_1, f_2, \ldots, f_n$, and the primary output. A test that detects the multiple fault $(f_1, f_2, \ldots, f_n)$ detects also fault f_a, since the two faults are equivalent. If T detects all such detectable multiple faults, then it also detects all detectable single faults in the network.

For the network in fig. 4.1 there are four undetectable faults on the checkpoints. Faults 1-0 and 2-0 belong to the same equivalence class, and so do faults 3-0 and 4-0. Choosing 1-0 and 3-0 to represent the respective equivalence classes, then the set U = (1-0,3-0). Fault 8-0 is equivalent to the multiple fault (1-0,3-0). A complete single fault test set for this network can be generated by considering all detectable single faults on the checkpoints and the multiple fault (1-0,3-0).

It should be noted that the result of lemma 4.1 also holds for all networks with fanout, where no primary input fans out. Fanout on the input should not be considered as true network fanout; however, it may be, in which case the result of lemma 4.1 does not hold as is illustrated by the following example. The network in fig. 4.2 has six checkpoints enumerated 1-6. Test set $(abc, \bar{a}\bar{b}c, a\bar{b}c, \bar{a}\bar{b}\bar{c}, \bar{a}b\bar{c})$ detects all detectable single faults on checkpoints.

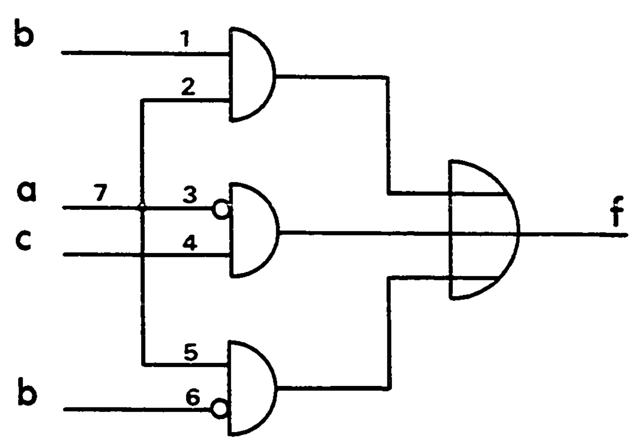


Figure 4.2 Network N4.2

Single faults 1-1,3-0 and 6-0 are not detectable and so is any combination of them. The above test set, however, does not detect fault 7-0 which is a detectable fault. Note that 7-0 is equivalent to the multiple fault (2-0,3-0,5-0). The undetectable fault 3-0 masks fault 2-0 under test abc, and it also masks 5-0 under test a$\bar{b}$c.

The previous discussion suggests that the following two approaches could be used to derive a single fault test set:

1. Consider single faults on all lines in the network and generate a test set that detects all such faults that are detectable.

2. Consider only faults on the checkpoints of the network, but derive a test set to detect single as well as multiple faults. Because any fault is equivalent to some single or multiple fault on the checkpoints, such a test set will detect all detectable faults within the network.

The second approach provides a solution to single and also multiple fault detection and it will therefore be employed in this thesis. A test set detecting all detectable single faults on the checkpoints (i.e. all single faults in D) can be derived by procedure 2.1. The set U of undetectable single faults on the checkpoints is obtained as a byproduct of this procedure, since if a fault p-0 (or p-1) is undetectable, there will be no 0 (or 1) entry in the p-th column of the checkpoint covering table. In the following section the problem of multiple fault detection will be examined.

4.2 Multiple Fault Detection

The presence of undetectable faults in redundant networks complicates the search for a multiple fault test set. Multiple faults in irredundant networks are of the same type, i.e., they all consist of detectable single faults. Whereas a multiple fault that occurs in a redundant network belongs to one of the following three classes:

Class 1: only undetectable single faults occuring simultaneously.

Class 2: only detectable single faults occuring simultaneously.

Class 3: undetectable and detectable single faults occuring simultaneously.

Multiple faults belonging to the above classes will now be analysed and a method that derives a minimal multiple fault test set will be described.

In order to find a minimal set of tests that detects all class 1 multiple faults that are detectable, it is necessary to consider all multiple faults consisting of two or more undetectable faults from the set U. If this set contains a class of equivalent faults, then the number of faults can be reduced by choosing only one fault to represent the class. A multiple fault $F = (f_1,f_2,\ldots,f_n)$, where $f_1,f_2,\ldots,f_n$ are undetectable faults, is detectable, if there exists at least one term X_i (or Y_j) which grows into X_i' (or Y_j') due to F and the intersection $X_i'.\bar{f}$ (or $Y_j'.f$) is nonempty. Note that the term X_i grows due to F if it does not disappear due to some fault $f_k \varepsilon F$ (see section 3.1).

Now, let us assume that a test set T has been derived, that detects all single faults from the set D and all detectable multiple faults consisting of the single faults in the set U. Then any detectable multiple fault that is not detected by T contains at least one fault $f_k \varepsilon D$. On the other hand if such a multiple fault is not detected, then f_k must be masked.

Let $F = (f_1,f_2,\ldots,f_n)$ be a multiple fault of class 2. Because all the individual faults $f_1,f_2,\ldots,f_n$ are detected by T, F is not detected by T only if every fault f_k, $k = 1,2,\ldots,n$, is masked by some subfault of F. The treatment of this type of multiple fault is the same as in the case of irredundant networks, and if such undetectable multiple faults exist, then the masking graph under test set T contains a strongly connected subgraph.

A multiple fault that belongs to class 3 consists of at least one detectable and at least one undetectable fault occuring simultaneously. Let $F = (f_{u1},\ldots,f_{um},f_{d1},\ldots,f_{dn})$ be such a fault, where f_{ui}, $i = 1,2,\ldots,m$ are undetectable faults, and f_{dj}, $j = 1,2,\ldots,n$, are detectable faults. A test set T detects every fault f_{dj} and also every detectable subfault of $(f_{u1}, f_{u2}, \ldots, f_{um})$. F will be not detected by T if every one of the detectable faults is masked. This will happen if at least one of the following two cases occurs:

1. The detectable faults mask each other. Then the masking graph under test set T will have at least one loop.

2. One or more undetectable faults mask some detectable faults, which, in turn, can mask other detectable faults. In this case, there will be at least one path in the masking graph, such that the initial node in the path corresponds to some undetectable fault.

Provided that the number of undetectable faults is not very large, the approach outlined above can be used to derive a minimal multiple fault test set with an effort comparable to that one required by irredundant networks.

Example 4.1: To illustrate, a multiple fault test set will be derived for the network in fig. 4.3. The ESP forms of the network are:

$$ESP(f) = a_{16}b_{26}c_3 + \bar{a}_{17}c_4 + \bar{a}_{17}d_5 + \bar{b}_{27}c_4 + \bar{b}_{27}d_5$$

$$ESP(\bar{f}) = \bar{a}_{16}\bar{c}_4\bar{d}_5 + \bar{b}_{26}\bar{c}_4\bar{d}_5 + a_{17}b_{27}\bar{c}_3 + \bar{c}_3\bar{c}_4\bar{d}_5 + + \bar{a}_{16}a_{17}b_{27} + a_{17}b_{27}\bar{b}_{26}$$

A minimal test set that detects all detectable single faults on checkpoints is derived by procedure 2.1. Table 4.1 and Table 4.2 are obtained by performing steps 2-5 of this procedure. From Table 4.2 test set T

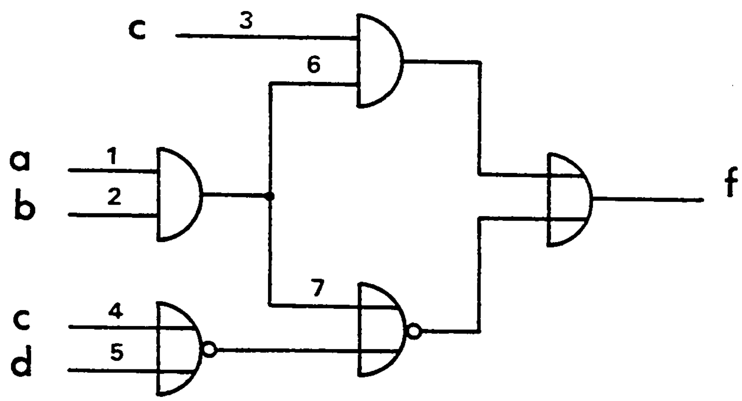


Figure 4.3 Network N4.3

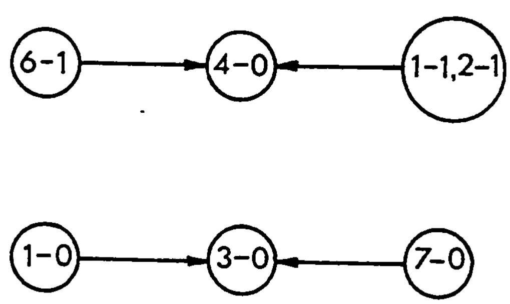


Figure 4.4 Masking graph for network N4.3

TABLE 4.1

Testing Table for Network N4.3

X_i	X_i'	$Z_k = X_i' . \bar{f}$	Covered by T
$a_{16}b_{26}c_3$	ab	$ab\bar{c}$	X
	ac	$\emptyset$	
	bc	$\emptyset$	
	c	$\emptyset$	
$\bar{a}_{17}c_4$	$\bar{a}$	$\bar{a}\bar{c}\bar{d}$	X
	c	$\emptyset$	
$\bar{a}_{17}d_5$	$\bar{a}$	$\bar{a}\bar{c}\bar{d}$	X
	d	$ab\bar{c}d$	X
$\bar{b}_{27}c_4$	$\bar{b}$	$\bar{b}\bar{c}\bar{d}$	X
	c	$\emptyset$	
$\bar{b}_{27}d_5$	$\bar{b}$	$\bar{b}\bar{c}\bar{d}$	X
	d	$ab\bar{c}d$	X

Y_j	Y_j'	$W_m = Y_j' . f$	Covered by T
$\bar{a}_{16}\bar{c}_4\bar{d}_5$	$\bar{a}\bar{c}$	$\bar{a}\bar{c}d$	X
	$\bar{a}\bar{d}$	$\bar{a}c\bar{d}$	X
	$\bar{c}\bar{d}$	$\emptyset$	
$\bar{b}_{26}\bar{c}_4\bar{d}_5$	$\bar{b}\bar{c}$	$\bar{b}\bar{c}d$	X
	$\bar{b}\bar{d}$	$\bar{b}c\bar{d}$	X
	$\bar{c}\bar{d}$	$\emptyset$	
$a_{17}b_{27}\bar{c}_3$	ab	abc	X
	$a\bar{c}$	$a\bar{b}\bar{c}d$	X
	$b\bar{c}$	$\bar{a}b\bar{c}d$	X
$\bar{c}_3\bar{c}_4\bar{d}_5$	$\bar{c}\bar{d}$	$\emptyset$	
	$\bar{c}\bar{c}$	$(\bar{a}+\bar{b})\bar{c}d$	X
$\bar{a}_{16}a_{17}b_{27}$	ab	abc	X
	$\bar{a}b$	$\bar{a}b(c+d)$	X
$a_{17}b_{27}\bar{b}_{26}$	ab	abc	X
	$a\bar{b}$	$a\bar{b}(c+d)$	X

$T = (ab\bar{c}d,\ \bar{a}\bar{b}\bar{c}\bar{d},\ a\bar{b}\bar{c}d,\ \bar{a}b\bar{c}d,\ \bar{a}\bar{b}c\bar{d},\ abcd)$ detects all detectable single faults on checkpoints. Fault 6-1 is the only fault that is not detectable, hence U = (6-1) and T also detects all class 1 multiple faults (there are not any). Next, the masking graph is constructed. Term $X_1 = a_{16}b_{26}c_3$ is not checked for growth due to literals a and b (see Table 4.1). For the term $X_1' = c_3$, the intersection

TABLE 4.2

Checkpoint Covering Table for Network N4.3

	1	2	3	4	5	6	7
$ab\bar{c}d$	0	0	1	-	-	-	0
$\bar{a}\bar{b}\bar{c}\bar{d}$	-	-	-	1	1	-	-
$a\bar{b}\bar{c}d$	-	1	-	-	0	-	-
$\bar{a}b\bar{c}d$	1	-	-	-	0	-	-
$\bar{a}\bar{b}c\bar{d}$	-	-	-	0	-	-	-
$abcd$	-	-	0	-	-	0	-

$X_1'.\bar{f}.T = \emptyset$ and the intersection $X_1'.\bar{X}_1'.f.T$ contains test $\bar{a}\bar{b}c\bar{d}$. This test detects fault 4-0 by checking for growth of $Y_1 = \bar{a}_{16}\bar{c}_4\bar{d}_5$ and $Y_2 = \bar{b}_{26}\bar{c}_4\bar{d}_5$. Since both terms disappear due to fault 6-1 or fault (1-1,2-1), 6-1/2-0 and (1-1,2-1)/4-0. The masking graph is in fig. 4.4, where the remaining masking relations have been obtained as a result of the unchecked growth of X_2 and X_4. In fig. 4.4, fault 1-0 is chosen to represent the equivalence class (1-0,2-0) and fault 3-0 represents the equivalence class (3-0,6-0). The masking graph does not have any cycles, but there is a path starting with the undetectable fault 6-1. This fault masks fault 4-0 and test set T should be enlarged to detect the multiple fault (6-1,4-0). Because no X_i term grows due to (6-1,4-0) into X_i' such that $X_i'.\bar{f} \neq \emptyset$, and the same is true for all Y_j terms, fault (6-1,4-0) is not detectable and test set T detects all detectable faults in the network. This example also shows that a detectable fault in a redundant network may become undetectable in the presence of some other undetectable fault.

It should be noted that the number of undetected multiple faults is larger when the paths in the masking graph are longer than those in fig. 4.4. For example, consider the masking graph in fig. 4.5, where f_1 is an undetectable fault and $f_2,f_3,\ldots,f_n$ are detectable faults. Any multiple fault $F = (f_k,f_{k+1},\ldots, f_{k+m})$, where $k > 1$, $m \leq n-k$, and any multiple fault $F = (f_1,f_2,\ldots, f_{k-2},f_k,f_{k+1},\ldots)$ is detected, since f_k is not masked if f_{k-1} does not occur. However, all the multiple faults (f_1,f_2), $(f_1,f_2,f_3),\ldots,$ $(f_1,f_2,\ldots,f_n)$ are not detected and the test set must be enlarged to detect all such faults that are detectable. In addition, provided that the masking graph has more than one path and/or loop, there are undetected multiple faults that correspond to combinations of these simple cases. Because no line in the network can be s-0 and s-1 at the same time, a limit is placed

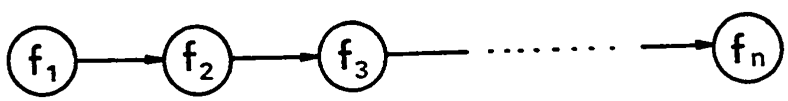


Figure 4.5 An example of a masking graph

on the number of such faults that must be practically considered.

The following procedure can be used to derive a minimal multiple fault test set for a redundant network.

Procedure 4.1:

1. Evaluate $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$.

2. For each X_i and Y_j term compute all terms $X_i^!$ and $Y_j^!$ that are one variable less. For each $X_i^!$ and $Y_j^!$ term compute $Z_k = X_i^!.\bar{f}$ and $W_m = Y_j^!.f$.

3. For any Z_k or W_m term that is empty, repeat step 2 with the corresponding $X_i^!$ or $Y_j^!$ term.

4. Find a minimal covering MC of all the Z_k and W_m terms generated above.

5. Construct the checkpoint covering table, where the column headings are the checkpoints and the row headings are the members of MC. The entries are filled in according to lemma 2.2,2.3 and 2.4. From this table find the set U of single undetectable faults. For each class of equivalent faults in this set, keep only one fault to represent the class.

6. Form the set G_1 of all class 1 multiple faults that consist of single faults in the set U. For each fault $F \in G_1$, add a new column to the checkpoint covering table and fill the entries in this column. If some fault is not detected by any one of the tests represented by the row

headings of the table, mark this fault as undetectable.

7. From this table, select a minimal test set T that detects all detectable single faults and all class 1 multiple faults that are detectable. If $T \equiv MC$, stop.

8. Use procedure 3.1 to construct the masking graph and from this graph find the set G_2 of all multiple faults that are not detected by T. For each fault $F \in G_2$ add a new column to the table and fill in the entries, but for only those rows that do not belong to T.

9. From the table select a minimal test set T' that detects all detectable faults from G_2. The multiple fault test set $T_m = T \cup T'$.

The minimal covering MC derived in step 4 of the above procedure represents a test set that weakly checks for growth of all X_i and Y_j terms. By theorem 3.4, such a test set detects all detectable faults in the network. Therefore, any detectable multiple fault considered in step 6 and 8 must be detected by some member of MC. Provided that MC is a minimal covering of the Z_k and W_m terms, then the test set T_m generated by the above procedure will be a minimal test set. However, there may be more than one minimal covering of the Z_k and W_m terms, and to prove the minimality of T_m it would be necessary to show that the length of T_m is independent of the choice of the minimal covering. This is rather difficult, but practice indicates that a minimal test set is generated in every case.

Example 4.2: Procedure 4.1 will be used to derive a test set for the network in fig. 4.6. The ESP forms of this network are

$ESP(f) = a_1b_2c_3d_4 + a_1b_2\bar{d}_6 + \bar{b}_5c_3d_4 + \bar{b}_5\bar{d}_6$

$ESP(\bar{f}) = \bar{a}_1b_5 + \bar{b}_2b_5 + \bar{c}_3d_6 + \bar{d}_4d_6$

TABLE 4.3

Checkpoint Covering Table for Network N4.6

	1	2	3	4	5	6
$\bar{a}bcd$	1	-	-	-	0	-
$ab\bar{c}d$	-	-	1	-	-	0
$\bar{a}bc\bar{d}$	1	-	-	-	0	-
$\bar{a}\bar{b}\bar{c}d$	-	-	1	-	-	0
$\bar{a}\bar{b}cd$	-	-	0	0	1	-
$ab\bar{c}\bar{d}$	0	0	-	-	-	1

The Z_k and W_m terms generated in step 2 are

Z_k: $ab\bar{c}d$, $\bar{a}bcd$, $\bar{a}b\bar{d}$, $\bar{b}\bar{c}d$

W_m: $\bar{a}\bar{b}(c+\bar{d})$, $ab(c+\bar{d})$, $\bar{b}(c+\bar{d})$, $(a+\bar{b})\bar{c}\bar{d}$, $(a+\bar{b})cd$, $(a+\bar{b})\bar{d}$

Step 3 produces two additional Z_k terms, $a\bar{c}d$ and $\bar{a}bc$. A minimal covering MC of all the Z_k and W_m terms above is

MC = ($\bar{a}bcd$, $ab\bar{c}d$, $\bar{a}bc\bar{d}$, $\bar{a}\bar{b}\bar{c}d$, $\bar{a}\bar{b}cd$, $ab\bar{c}\bar{d}$)

Lemma 2.2 and 2.3 are used to fill the entries in the checkpoint covering table (Table 4.3). From this table it is seen that faults 2-1 and 4-1 are undetectable. The only term which grows due to both 2-1 and 4-1 is the term $X_1 = a_1b_2c_3d_4$. However, for the term $X_1' = a_1c_3$, the

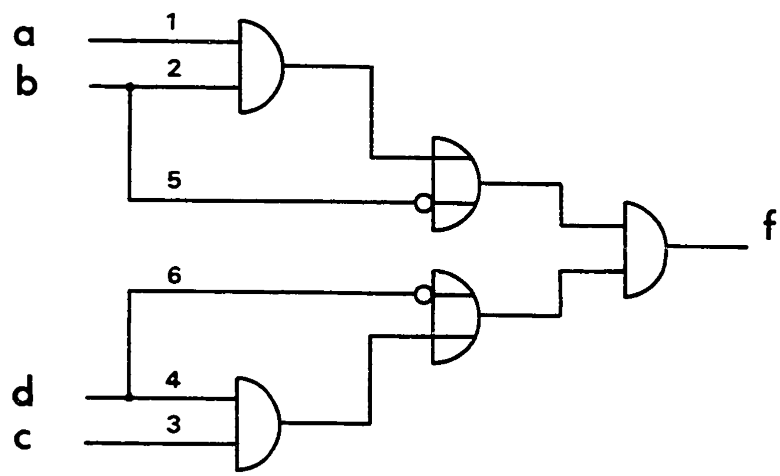


Figure 4.6 Network N4.6

intersection $ac.\bar{\bar{f}} = \emptyset$. Hence the fault (2-1,4-1) is undetectable. The following minimal test set T is selected from Table 4.3

$$T = (\bar{a}bc\bar{d},\ \bar{a}\bar{b}\bar{c}d,\ \bar{a}\bar{b}cd,\ ab\bar{c}\bar{d})$$

Procedure 3.1 produces only one masking relation and this is (3-1,4-1)/6-1. This relation has been obtained as a result of the unchecked growth of the term $X_1 = a_1b_2c_3d_4$. Because fault 3-1 is not masked, there is no need to enlarge T and this test set is also a minimal multiple fault test set.

The inevitable price paid for the minimal test set generated by procedure 4.1 is the increased complexity of the computations involved, specifically when the number of undetectable faults is large. The minimal covering MC computed in step 4 of this procedure corresponds to the test set derived by the method of Bossen and Hong [5]. Obviously, the effort required to obtain such a test set is considerably smaller. On the other hand, there is a difference in the length of the test set generated. Denoting by ℓ the length of a minimal test set T_m as derived by procedure 4.1 or by procedure 3.2, and by ℓ' the length of the test set generated by procedure G[5], then the following comparison can be made:

network N2.1	$\ell = 5, \ell' = 9$
network N4.2	$\ell = 6, \ell' = 6$
network N4.5	$\ell = 4, \ell' = 6$
network N3.4	$\ell = 6, \ell' = 9$

From which the conclusion can be made that while the procedure is more complex it derives a smaller test set for some examples.

4.3 Concluding Remarks

It should be pointed out that in redundant networks the problem of multiple fault detection is more important than in irredundant networks. It is often argued, that detection of single faults is sufficient provided that the time span between testing is short. This may be true for irredundant networks, but it is not true for the redundant ones. For example, let N_1 be an irredundant network, N_2 a redundant network, and T_1 and T_2 a single fault test set for N_1 and N_2, respectively. For N_1 the following sequence of events (f_1 occurs, T_1 applied (f_1 removed), f_2 occurs, T_1 applied) results in detecting fault f_1 as well as fault f_2. However, if f_1 is an undetectable fault that masks some detectable fault f_2 under test set T_2 in network N_2, then after the sequence

(f_1 occurs, T_2 applied, f_2 occurs, T_s applied)

N_2 may no longer operate correctly, but the network passes the test both times.

It is felt that for this reason the problem of fault detection in redundant networks should never be restricted to only single faults.

Chapter 5

FAULT DETECTION IN SPECIAL NETWORKS

The test generation methods presented in chapters 2-4 employ both the ESP(f) and ESP($\bar{f}$) form when deriving the test set. In this chapter, a method which uses only one of the two forms to generate a test set for networks with equal inversion parity of all reconverging paths will be described. Some additional results for fanout-free networks are given in section 5.2, and an alternative way of generating tests is shown in section 5.3. Finally, the problem of fault testing in multiple output networks will be discussed.

5.1 Networks With Equal Inversion Parity of all Reconverging Paths

The set of networks with equal inversion parity of all reconverging paths (denoted networks with EIP) consists of all networks where the number of inverters, modulo 2, along any two reconverging paths is the same. In such networks any fault causes the growth of either X_i term(s) or Y_j term(s). This is also true for any fault in a fanout-free network and from this point of view the set of all fanout-free networks may be considered as a proper subset of the set of networks with EIP. If the ESP form of a network with EIP contains a term a_pW, then it cannot contain a term $\bar{a}_pU$. Thus, the set of functions realized by networks with EIP is a subset of unate functions.

5.1.1 Single Faults

Because the inversion parity of all paths connecting some checkpoint and the output of the network is the same, no fault causes both the disappearance of an X_i (or Y_j) term and the growth of another X_i (or Y_j) term at the same time. Consequently, if a test checks for the presence of a term, then it detects all faults that cause the disappearance of the same term. Lemma 5.1 states this result formally.

Lemma 5.1: Let N be a normal network with $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$. If a test t checks for the presence of the term $a_{ij\ldots l}\bar{b}_{pq\ldots s}W$ and the inversion parity of all reconverging paths in N is the same, then t detects the following faults:

- s-0 and s-1 faults on checkpoints i and p, respectively,
- s-0 and s-1 faults on checkpoints k and r, respectively, provided that the inversion parity along the path i,j,...,k (p,q,...,r) is even,
- s-1 and s-0 faults on checkpoints k and r, respectively, provided that the inversion parity along the path i,j,...,k (p,q,...,r) is odd.

Test t also detects faults on all checkpoints associated with literals in W. The type of fault is determined as described above.

<u>Proof</u>: If test t checks for the presence of some term, say X_α, then t is contained in the intersection $X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i$ and the output of the fault-free network under input t is 1. If a fault f_1 occurs, such that $f_1 : X_\alpha \rightarrow \emptyset$, then the output of the network is 0 because t is not covered by any X_i term, and no X_i term can grow due to f_1 because of the EIP property. Hence t detects any fault that causes the disappearance of X_α. It was shown in chapter 2, that for a normal network such faults can be determined from the type of literal (primed or unprimed) and the inversion parity along the particular path.

If a network has an irredundant ESP form, then it is possible to check for the presence of every X_i or every Y_j term. A minimal test set for such networks with EIP can be generated while using only one of the two forms. The following procedure summarises the necessary steps.

<u>Procedure 5.1</u>:

1. Evaluate either $ESP(f) = \Sigma X_i$ or $ESP(\bar{f}) = \Sigma Y_j$. (In steps 2 to 6, it is assumed that ESP(f) is used).

2. For every term X_i generate all terms X_i' that are one variable less. For each X_i' compute $Z_k = X_i' . \bar{f}$.

3. If there exists some X_i term having more than one Z_k term that is empty, then repeat step 2 with those X_i' terms having Z_k terms that are empty and common checkpoints, by by removing the variables with common checkpoints according

to lemma 2.4.

4. For each term X_α compute $W_m = X_\alpha . \prod_{i \neq \alpha} \bar{X}_i$.

5. Find a minimal covering MC of all the Z_k and W_m terms generated above.

6. Construct the checkpoint covering table where the column headings are the checkpoints and the row headings are the members of MC. The entries for rows that cover the Z_k terms are filled according to lemmas 2.2,2.3, and 2.4, and the entries for rows that cover the W_m terms according to lemma 5.1.

7. From the checkpoint covering table select T_s as a minimal test set detecting all s-0 and s-1 faults on checkpoints.

The above procedure generates a minimal single fault test set for any network having either an ESP(f) or ESP($\bar{f}$) form that is irredundant. The minimality and completeness of the test set with respect to faults that cause growth of X_i terms follows from the proof of procedure 2.1. Provided that the ESP(f) form is irredundant, it is possible to check for the presence of every X_i term and therefore a test will be generated for any fault that causes some X_i to disappear. A test t_1 that checks for the presence of the term X_α detects all faults that cause X_α to disappear. A test t_2, such that $t_2 \in X_\alpha . X_\beta . \prod_{i \neq \alpha,\beta} \bar{X}_i$ detects only those faults causing disappearance of X_α as well as X_β, i.e., faults on only those checkpoints common to X_α and X_β. Thus, by generating tests that detect the largest number of faults it is ensured that a test set with a minimal number of tests will be derived.

A complete test set for networks that do not have at least one ESP form that is irredundant cannot be, in general, derived by the above procedure. However, if every checkpoint is associated with at least one term X_α such that $X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i \neq \emptyset$, then the above procedure can be applied and it will produce a minimal test set. This is illustrated by the following example.

<u>Example 5.1</u>: For the network in fig. 5.1 the ESP(f) form is

$$ESP(f) = a_{17}c_5 + b_{27}c_5 + a_{18}d_6 + b_{28}d_6 + \bar{b}_3\bar{d}_4$$

Steps 2-4 produce table 5.1 and a minimal covering of all the Z_k and W_m terms is

$$MC = (ab\bar{c}\bar{d},\ \bar{a}\bar{b}cd,\ \bar{a}bcd,\ a\bar{b}\bar{c}d,\ \bar{a}b\bar{c}d,\ \bar{a}\bar{b}\bar{c}\bar{d})$$

From table 5.2 a minimal single fault test set T_s is

$$T_s = (ab\bar{c}\bar{d},\ \bar{a}\bar{b}cd,\ \bar{a}bc\bar{d},\ a\bar{b}\bar{c}d,\ \bar{a}\bar{b}\bar{c}\bar{d})$$

Because T_s checks strongly for the growth of all X_i terms, it also detects all multiple faults.

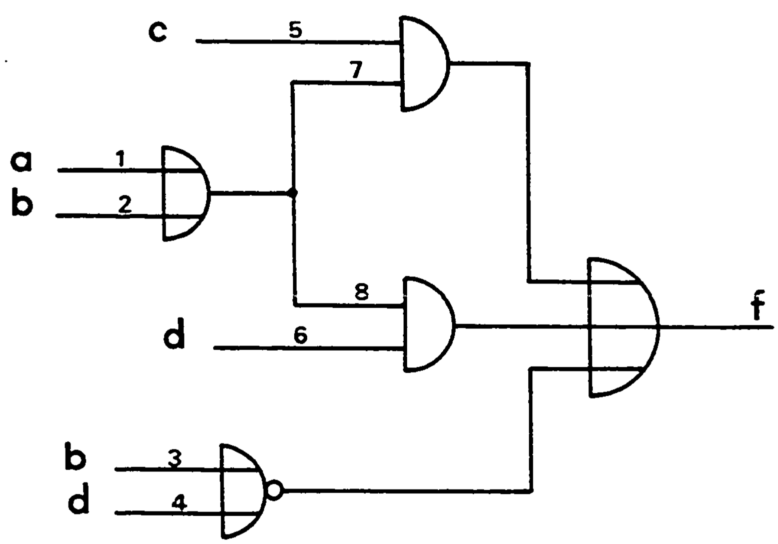


Figure 5.1 Network N5.1

TABLE 5.1

Testing Table for Network N5.1

X_i	X'_i	$Z_k = X'_i \cdot \bar{f}$	$W_m = X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i$
$a_{17}c_5$	a c	$ab\bar{c}\bar{d}$ $\bar{a}\bar{b}cd$	$\emptyset$
$b_{27}c_5$	b c	$\bar{b}cd$ $\bar{a}\bar{b}cd$	$\bar{a}bc\bar{d}$
$a_{18}d_6$	a d	$ab\bar{c}\bar{d}$ $\bar{a}\bar{b}d$	$a\bar{b}\bar{c}d$
$b_{28}d_6$	b d	$\bar{b}cd$ $\bar{a}\bar{b}d$	$\bar{a}b\bar{c}d$
$\bar{b}_3\bar{d}_4$	$\bar{b}$ $\bar{d}$	$\bar{a}\bar{b}d$ $b\bar{c}\bar{d}$	$(\bar{a}+\bar{c})\bar{b}\bar{d}$

TABLE 5.2

Checkpoint Covering Table for Network N5.1

	1	2	3	4	5	6	7	8
$ab\bar{c}\bar{d}$	–	–	0	–	1	1	–	–
$\bar{a}\bar{b}cd$	1	1	–	0	–	–	1	1
$\bar{a}bc\bar{d}$	–	0	–	–	0	–	0	–
$a\bar{b}\bar{c}d$	0	–	–	–	–	0	–	0
$\bar{a}b\bar{c}d$	–	0	–	–	–	0	–	0
$\bar{a}\bar{b}\bar{c}\bar{d}$	–	–	1	1	–	–	–	–

5.1.2 Multiple Faults

In the preceding section it was shown how to derive a minimal single fault test set for networks with EIP from only one of the two ESP forms. In general, it is difficult to perform a complete multiple fault analysis without using both the ESP(f) and ESP($\bar{f}$). Provided that is is possible to strongly check for growth of all terms of the particular ESP form being used, multiple fault analysis can be avoided and a nearly minimal multiple fault test set can be derived directly. In the special case of a network with EIP, it is sufficient to fulfill conditions that are less stringent.

The set of all single faults in a network with EIP is partitioned into two blocks A and B:

A = set of all faults causing growth of X_i terms

B = set of all faults causing growth of Y_j terms.

It follows from theorem 3.2 that any fault $f_a \in A$ can be masked only by some fault $f_b \in B$, and vice versa. Now, suppose that a test set T_s detects all single faults in a network N with ESP(f) = ΣX_i and ESP($\bar{f}$) = ΣY_j. If some multiple fault F exists, that is not detected by T_s, F must contain at least one fault $f_a \in A$ and at least one fault $f_b \in B$, since two or more faults belonging to the same set cannot mask each other. Consequently, there must exist at least one term X_i that grows into X_i' where

$$X_i^{!}.\bar{f}.T_s = \emptyset$$

and $$X_i^{!}.\bar{X}_i.f.\ T_s \neq \emptyset \qquad (5.1)$$

and at least one term Y_j that grows into $Y_j^{!}$ where

$$Y_j^{!}.f.T_s = \emptyset$$

and $$Y_j^{!}.\bar{Y}.\bar{f}.\ T_s \neq \emptyset \qquad (5.2)$$

The test set T_s detects all multiple faults if either (5.1) does not hold for any X_i term, or if (5.2) does not hold for any Y_j term. Note that (5.1) or (5.2) does not hold if at least one of the two equations is not satisfied.

The above fact can be exploited when a multiple fault test set is to be derived. As an example, consider the network in fig. 5.2, where

$ESP(f) = a_{17}b_{27}c_3d_4 + a_{18}b_{28}\bar{c}_5\bar{d}_6$

Test set

$T_s = (abcd,\ ab\bar{c}\bar{d},\ \bar{a}bcd,\ a\bar{b}\bar{c}\bar{d},\ ab\bar{c}d,\ abc\bar{d})$

is a minimal single fault test set. T_s does not check for growth of X_1 due to variable b, and for growth of X_2 due to variable a. However the intersections

$$X_1^{!}.\bar{X}_1.f.T_s = a\bar{b}cd(abcd + ab\bar{c}\bar{d}) = \emptyset$$

$$X_2^{!}.\bar{X}_2.f.T_s = \bar{a}b\bar{c}\bar{d}(abcd + ab\bar{c}\bar{d}) = \emptyset$$

are empty and hence T_s also detects all multiple faults.

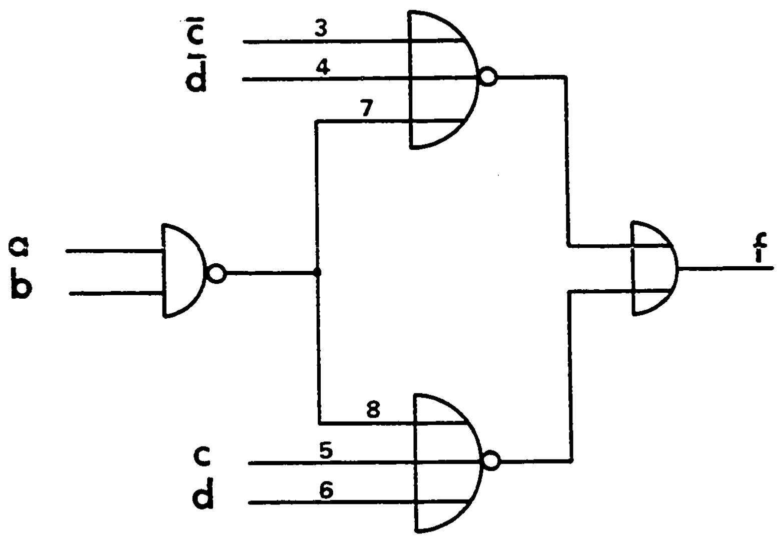


Figure 5.2 Network N5.2

5.2 Multiple Faults in Fanout-Free Networks

In section 3.1 an example was given, where a single fault test set for a fanout-free network does not detect a multiple fault of multiplicity five. Similarly, for the network in fig. 5.3, the test set T_s

$T_s = (\bar{a}bcd\bar{e}\bar{g},\ a\bar{b}\bar{c}d\bar{e}g,\ abc\bar{d}\bar{e}g,\ a\bar{b}cde\bar{g},\ ab\bar{c}\bar{d}\bar{e}g,\ \bar{a}bc\bar{d}e\bar{g})$

does not detect the multiple fault (2-1,4-1,5-0,6-0). The multiplicity of this fault is four. However, it will now be shown that any single fault test set for a fanout-free network detects all multiple faults of multiplicity two and three.

<u>Lemma 5.2</u>: Let N be a fanout-free network with $ESP(f) = \Sigma X_i$ and $ESP(\bar{f}) = \Sigma Y_j$; then no two checkpoints appear together within both a single X_i and a single Y_j term.

<u>Proof</u>: For a one level network realized by a single NAND gate with p inputs $h_1, h_2, \ldots, h_p$, the ESP forms are

$$ESP(f) = \bar{h}_1 + \bar{h}_2 + \ldots + \bar{h}_p \tag{5.3}$$

$$ESP(\bar{f}) = h_1 . h_2 . \ldots . h_p \tag{5.4}$$

Without loss of generality, the above equations can be used to represent the ESP forms of any one level network, since for any AND, OR, or NOR gate with p inputs the ESP forms are similar to (5.3) and (5.4). The lemma can now be proved by induction.

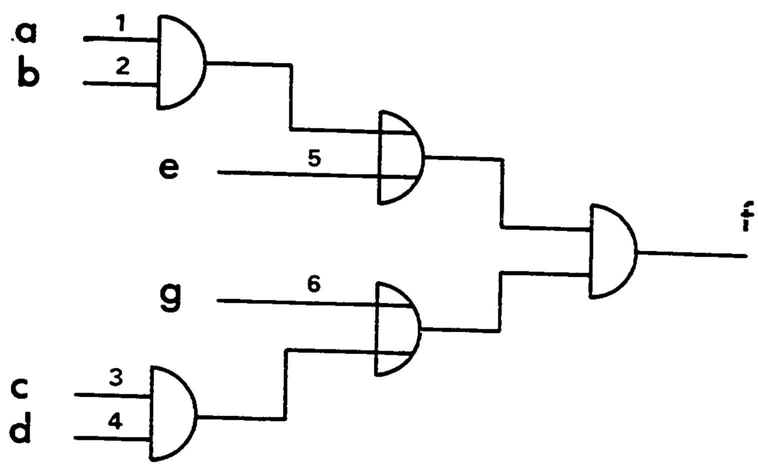


Figure 5.3 Network N5.3

(a) Clearly, for any one level network no two checkpoints appear within both a single X_i and a single Y_j term.

(b) Let N be an n-level fanout-free network with r checkpoints 1,2,...,r, and let the ESP forms of N be

$$ESP(f) = \Sigma X_i \qquad (5.5)$$

$$ESP(\bar{f}) = \Sigma Y_j \qquad (5.6)$$

By assumption, no two checkpoints of N appear within both a single X_i and a single Y_j term in (5.5) and (5.6). Let N' be an (n+1)-level network obtained from N by replacing the r checkpoint variables with r gates $G_1, G_2, \ldots, G_r$. Let ESP(g_k) and ESP($\bar{g}_k$) be the ESP forms of the network corresponding to gate G_k, for k = 1,2,...,r. The ESP forms for the network N' are obtained by substituting ESP(g_k) and ESP($\bar{g}_k$), for k = 1,2,...,r, into (5.5) and (5.6). It follows from equations (5.3) and (5.4) that if two checkpoints appear within both a single X_i and a single Y_j term, that they must have done so in (5.3) and (5.4), which is a contradiction. Hence no two checkpoints appear within both a single X_i term and a single Y_j term.

It should be noted that the result of lemma 5.2 may not hold for networks that contain other types of logical gates than AND, OR, NAND, NOR and NOT. For example, for an Exclusive-OR gate with two inputs, a and b, the ESP forms are

$$ESP(f) = a_1\bar{b}_2 + \bar{a}_1 b_2$$

$$ESP(\bar{f}) = a_1 b_2 + \bar{a}_1\bar{b}_2$$

Here, two checkpoints appear within both a single X_i and a single Y_j term. Networks which contain such more complex logical modules should not be considered as fanout-free, since a practical realization of the module is likely to contain fanout (see section 1.1). The following theorem has been proved by Hayes [18].

Theorem 5.1: Any single fault test set T_s for a fanout-free network also detects all multiple faults of multiplicity two and three.

Proof: The theorem will be proved by showing that no two or three single faults can form a closed masking graph.
(a) Consider the multiple fault $F = (f_1,f_2)$. Let fault f_1 be detected by a test t which checks for growth due to f_1 of some X_i term, say X_1. In the course of this proof we shall simply write $f_1:X_1 \rightarrow X_1'$ instead of "fault f_1 is detected by test $t \in T_s$ that is contained in the intersection $X_1'.\bar{f}$, where $f_1:X_1 \rightarrow X_1'$".
If f_2/f_1, then f_2: $X_1 \rightarrow 0$, consequently f_2: $Y_2 \rightarrow Y_2'$
If f_1/f_2, then f_1: $Y_2 \rightarrow 0$.
Fault f_1 cannot mask fault f_2 however, because f_1 and f_2 are faults on two different checkpoints and they cannot appear together within both X_1 and Y_2 (lemma 5.2).

(b) Consider the multiple fault $F = (f_1,f_2,f_3)$

Let f_1: $X_1 \rightarrow X_1'$

If f_2/f_1, then f_2: $X_1 \rightarrow 0$, and f_2: $Y_2 \rightarrow Y_2'$

If f_3/f_2, then f_3: $Y_2 \rightarrow 0$, and f_3: $X_3 \rightarrow X_3'$

If f_1/f_3, then f_1: $X_3 \rightarrow 0$, and f_1: $Y_1 \rightarrow Y_1'$

The last statement is a contradiction, because no single fault in a fanout-free network can cause growth of some X_i as well as some Y_j term. Since f_1 causes growth of X_1, f_1 cannot mask f_3. Hence, if test set T_s detects all single faults it also detects all multiple faults of multiplicity two and three.

A single fault test set T_s for a fanout-free network also detects a large portion of multiple faults of multiplicity greater than three. If two or more single faults occur that belong to the same class of equivalent faults, then all such faults can be represented by only one fault. Hence T_s will also detect any multiple fault $F = (f_1,f_2,\ldots,f_n)$ such that faults $f_1,f_2,\ldots,f_n$ do not belong to more than three different equivalence classes.

5.3 Test Generation for Large Circuits

For large circuits, the number of terms in the ESP form may become very high. Thus, although the use of the ESP forms for test generation is desirable, specifically for the purpose of multiple fault detection, the test generation methods described earlier may not be feasible when the network is very large. Also, it is often required to derive tests for only certain faults, or to find all tests which detect a particular fault. A method that can be used in such cases will now be described.

We denote by p-d the fault "checkpoint p is stuck at d", where d is 0 or 1, but not both at the same time. A test t detects fault p-d, if, under input t, the output of the correct network is different from the output of the network in the presence of the fault p-d. Let f(p-d) denote the output function realized by the network in the presence of fault p-d. Then the set T(p-d) of all tests detecting fault p-d is:

$$T(p\text{-}d) = f(p\text{-}d) \oplus f,$$

where f is the output of the correct network.

Let Z be the Boolean expression describing the function realized by the network. For each checkpoint of the network, there is a subscript attached in the corresponding position in the expression. For example, for the network in fig. 5.4 this expression is

$$Z = \overline{(a_1\bar{b}_2\bar{c}_3 \cdot (\overline{c_4\bar{d}_5e_6})_{10})} \cdot \overline{((\overline{c_4\bar{d}_5e_6})_{11} \cdot (b_7\bar{c}_8)_{12})} + (b_7\bar{c}_8)_{13} \cdot e_9$$

Since the logical value on checkpoint p is the presence of fault p-d is d regardless of the value of the variable(s) applied, f(p-d) can be obtained from Z by substituting d for any literal or subexpression that is associated with subscript p. In contrast to the evaluation of the ESP form, all the Boolean identities can be used when computing f(p-d) from Z. It should be noted that when the notation above is used, the range of a NOT operator may not be defined properly. For example, when computing f(10-0), it may not be clear whether to substitute 0 for the expression $\overline{c_4\bar{d}_5e_6}$ or for the expression $c_4\bar{d}_5e_6$. This can be solved by using a special symbol (e.g., "¬") and by including parenthesis to define the range of every operator. The previous notation is satisfactory, however, provided that a diagram of the network is used at the same time. This latter solution will be employed herein.

Assuming that f is the function realized by the network, then the set T(p-d) of all tests that detect fault p-d can be computed in the following two steps:

Procedure 5.2:

1. Compute f(p-d) by substituting d into Z for any literal or subexpression that is associated with subscript p. When evaluating f(p-d), disregard all subscripts and use all Boolean identities.

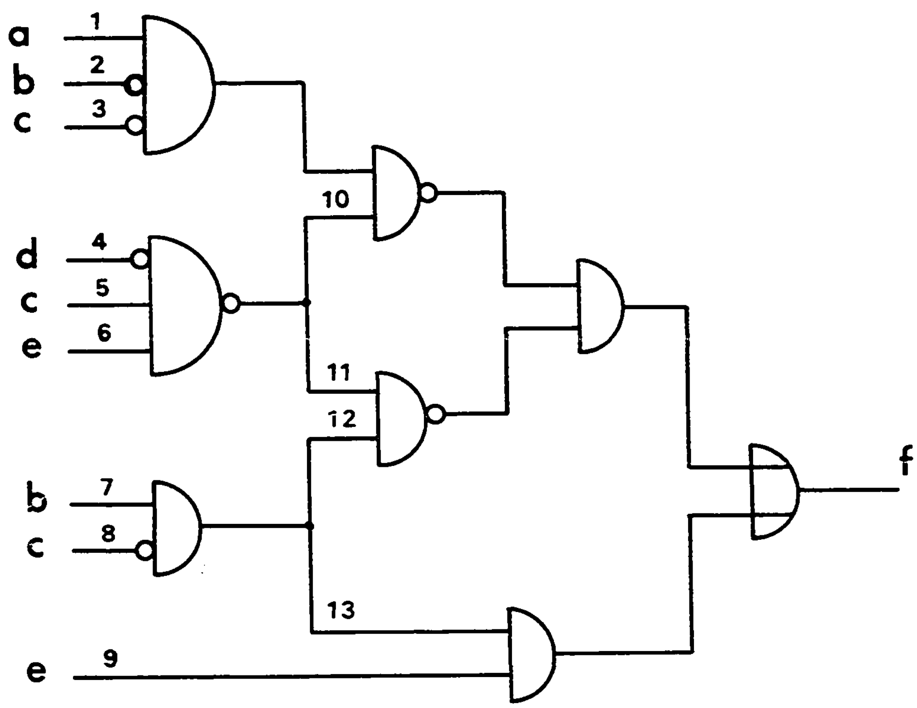


Figure 5.4 Network N5.4

2. Compute T(p-d) from the formula:

   $T(p\text{-}d) = f(p\text{-}d) \oplus f.$

A nearly minimal, or a minimal single fault test set for an irredundant network is obtained by:

- form the set S of all s-0 and s-1 fault test checkpoints, such that every class of equivalent faults is represented in this set by only one fault.
- compute $T(f_a)$ for every fault $f_a \in S$
- find a nearly minimal, or a minimal covering of all the T sets computed above.

The extension of the above method to redundant networks is straightforward and it follows from the discussion in section 4.1.

<u>Example 5.2</u>: The network in fig. 5.4 realizes the function $f = \bar{a}\bar{b} + c + b\bar{c}e$, and the output expression Z for for the network is

$$Z = \overline{(a_1\bar{b}_2\bar{c}_3 \cdot \overline{(c_4\bar{d}_5e_6)}_{10}}) \cdot (\overline{\overline{(c_4\bar{d}_5e_6)}_{11} \cdot (b_7\bar{c}_8)_{12}} + (b_7\bar{c}_8)_{12} \cdot e_9.$$

To compute T(1-1) evaluate f(1-1)

$$f(1\text{-}1) = \overline{(1.\bar{b}\bar{c}.\overline{c\bar{d}e}}).(\overline{\overline{cde}.b\bar{c}}) + b\bar{c}e =$$

$$= (b+c+c\bar{d}e).(c\bar{d}e + \bar{b}+c) + b\bar{c}e = c + b\bar{c}e.$$

Hence $T(1\text{-}1) = (c + b\bar{c}\bar{e}) \oplus (\bar{a}\bar{b} + c + b\bar{c}e) = \bar{a}\bar{b}\bar{c}$

Similarly, in the presence of the multiple fault (4-1,11-1) the function realized is

$$f(4\text{-}1,11\text{-}1) = \overline{(a\bar{b}\bar{c}.\overline{(1.\bar{d}e)})} \, . \, \overline{(1.b\bar{c})} + b\bar{c}e =$$

$$= \bar{a}\bar{b} + c + \bar{b}\bar{d}e + b\bar{c}e$$

$$T(4\text{-}1,11\text{-}1) = (\bar{a}\bar{b} + c + \bar{b}\bar{d}e + b\bar{c}e) \oplus (\bar{a}\bar{b} + c + b\bar{c}e) =$$

$$= a\bar{b}\bar{c}\bar{d}e$$

Although the above method can be used to generate tests for specific multiple faults, it is more difficult to generate a complete multiple fault test set, unless a large number of multiple faults is considered. Relative to the D-algorithm [41], this method will generate a test for every detectable fault with a computational effort that compares favourably. In addition, provided that a minimal covering of the T sets is selected, the resulting test set is minimal.

5.4 Multiple Output Networks

The problem of fault detection in multiple output networks is not substantially different from that in networks with a single output. A test set for a multiple output network can be obtained by generating a test set for the circuitry associated with each output separately and then taking the union of all such tests as the test set for the multiple output network. However, the test set derived in this way is not likely to be minimal. In order to obtain a minimal test set, all the outputs must be considered at the same time.

The <u>ESP form</u> of a <u>multiple output network</u> is the union of the ESP forms for every individual output. The test generation methods described earlier can be used for multiple outputs without change, provided that they are applied to this "composite" ESP form. This is demonstrated by the following example.

<u>Example 5.3</u>: The network in fig. 5.5 realizes two functions, g and h, where $g = abcd$ and $h = cd + \bar{b}\bar{c}$. The ESP forms are

$$ESP(g) = a_1 b_2 c_{5,9} d_{6,9} + \bar{b}_{3,7} \bar{c}_{4,7} c_{5,9} d_{6,9}$$
$$ESP(\bar{g}) = \bar{a}_1 b_{3,7} + \bar{a}_1 c_{4,7} + \bar{b}_2 b_{3,7} + \bar{b}_2 c_{4,7} + \bar{c}_{5,9} + \bar{d}_{6,9}$$
$$ESP(h) = c_{5,10} d_{6,10} + \bar{b}_{3,8} \bar{c}_{4,8}$$
$$ESP(\bar{h}) = \bar{c}_{5,10} b_{3,8} + \bar{c}_{5,10} c_{4,8} + b_{3,8} \bar{d}_{6,10} + c_{4,8} \bar{d}_{6,10}$$

To generate a minimal single fault test set, procedure 2.1 is applied to the ESP forms above. Steps 2-4 of this procedure produce Table 5.3. From this table, a minimal covering MC of all the Z_k and W_m terms is

$$MC = (abc\bar{d},\ ab\bar{c}d,\ a\bar{b}cd,\ \bar{a}bcd,\ abcd,\ a\bar{b}c\bar{d},\ \bar{a}\bar{b}\bar{c}\bar{d},\ \bar{a}\bar{b}\bar{c}d)$$

After the checkpoint covering table (Table 5.4) has been filled in, it is discovered that the network is redundant, because fault 7-1 is not detectable. The test set

$$T_s = (abc\bar{d},\ ab\bar{c}d,\ a\bar{b}cd,\ \bar{a}bcd,\ abcd,\ \bar{a}\bar{b}\bar{c}\bar{d})$$

is a minimal test set that detects all single detectable faults. Procedure 3.1 is now applied to determine whether

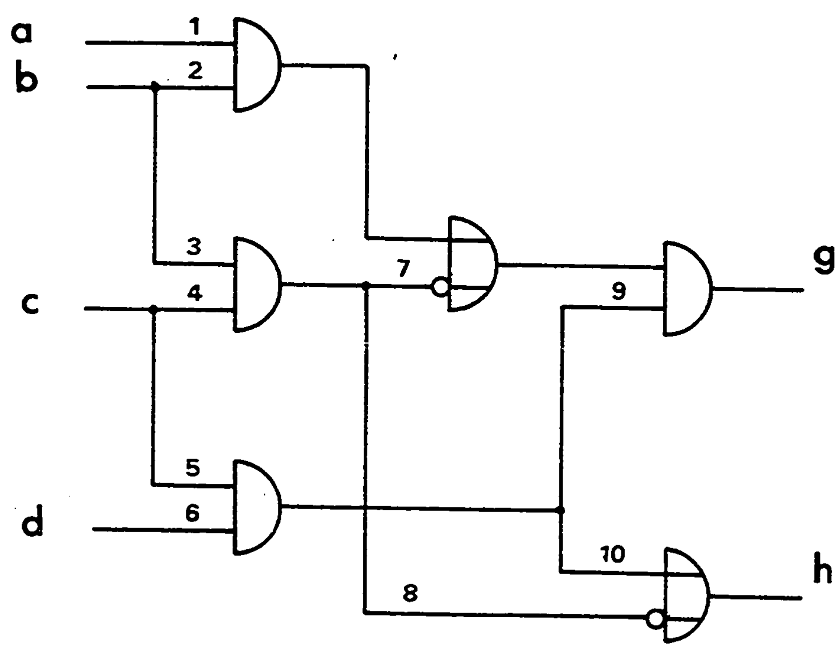


Figure 5.5 Network N5.5

TABLE 5.3

Testing Table for Network N5.5

X_i	X'_i	$Z_k=X'_i\cdot\bar{g}$	Covered by T_s
$a_1b_2c_{5,9}d_{6,9}$	abc	$abc\bar{d}$	X
	abd	$ab\bar{c}d$	X
	acd	$a\bar{b}cd$	X
	bcd	$\bar{a}bcd$	X
$\bar{b}_{3,7}\bar{c}_{4,7}c_{5,9}d_{6,9}$	$\bar{b}cd$	$\bar{b}cd$	X
	$\bar{b}\bar{c}d$	$\bar{b}\bar{c}d$	

Y_j	Y'_j	$W_m=Y'_j\cdot g$	Covered by T_s
$\bar{a}_1b_{3,7}$	$\bar{a}$	$\emptyset$	
	b	abcd	X
$\bar{a}_1c_{4,7}$	$\bar{a}$	$\emptyset$	
	c	abcd	X
$\bar{b}_2b_{3,7}$	$\bar{b}$	$\emptyset$	
	b	abcd	X
$\bar{b}_2c_{4,7}$	$\bar{b}$	$\emptyset$	
	c	abcd	X
$\bar{c}_{5,9}$	1	abcd	X
$\bar{d}_{6,9}$	1	abcd	X

TABLE 5.3 (Continued)

X_i	X_i'	$Z_k=X_i'.\bar{h}$	Covered by T_s
$c_{5,10}d_{6,10}$	c d	$c\bar{d}$ $b\bar{c}d$	X X
$\bar{b}_{3,8}\bar{c}_{4,8}$	$\bar{b}$ $\bar{c}$	$\bar{b}c\bar{d}$ $b\bar{c}$	 X
Y_j	Y_j'	$W_m=Y_j'.h$	Covered by T_s
$\bar{c}_{5,10}b_{3,8}$	$\bar{c}$ b	$\bar{b}\bar{c}$ bcd	X X
$\bar{c}_{5,10}c_{4,8}$	$\bar{c}$ c	$\bar{b}\bar{c}$ cd	X X
$b_{3,8}\bar{d}_{6,10}$	b $\bar{d}$	bcd $\bar{b}\bar{c}\bar{d}$	X X
$c_{4,8}\bar{d}_{6,10}$	c $\bar{d}$	cd $\bar{b}\bar{c}\bar{d}$	X X

there are any multiple faults that are not detected by T_s.

TABLE 5.4

Checkpoint Covering Table for Network N5.5

	1	2	3	4	5	6	7	8	9	10	
$abc\bar{d}$	-	-	-	-	-	1	-	-	1	1	X
$ab\bar{c}d$	-	-	0	-	1	-	-	0	1	-	X
$a\bar{b}cd$	-	1	-	0	0	0	0	-	-	0	X
$\bar{a}bcd$	1	-	-	-	0	0	-	-	-	0	X
$abcd$	0	0	-	-	0	0	-	-	0	0	X
$a\bar{b}c\bar{d}$	-	-	-	0	-	-	-	0	-	-	
$\bar{a}\bar{b}\bar{c}\bar{d}$	-	-	1	1	-	-	-	1	1	-	X
$\bar{a}\bar{b}\bar{c}d$	-	-	-	-	1	-	-	-	1	-	

Term $Y_3 = \bar{b}_2 b_{3,7}$ (of the ESP($\bar{g}$) form) is not checked for growth due to literal b. The intersection P is

$$P = Y_3^{!} . \bar{Y}_3 . \bar{g} . T_s = a\bar{b}cd$$

Test $a\bar{b}cd$ detects faults 2-1,4-0,5-0,6-0,7-0 and 10-0. The only masking relation produced is 7-1/4-0, since all the remaining faults are detected by checking for the growth of terms that do not disappear due to fault 7-1 or 3-1. Because of the unchecked growth of the term $X_2 = \bar{b}_{3,8}\bar{c}_{4,8}$ (of the ESP(h) form) it is also found that 4-0/6-0 and 4-0/10-0 under test $a\bar{b}cd$; however, both 6-0 and 10-0 are detected by two additional tests. Test set T_s is now enlarged to detect the multiple fault (7-1,4-0).

Term $\bar{b}_{3,8}\bar{c}_{4,8}$ grows due to this fault into $\bar{b}_{3,8}$ where

$$\bar{b}.\bar{h} = \bar{b}(b\bar{c} + c\bar{d}) = \bar{b}c\bar{d}$$

Thus, test set

$T_m = (abc\bar{d},\ ab\bar{c}d,\ a\bar{b}cd,\ \bar{a}bcd,\ abcd,\ \bar{a}\bar{b}\bar{c}\bar{d},\ \bar{b}c\bar{d})$

is a minimal test set that detects all detectable faults in this network.

Chapter 6

SOME BOUNDS ON THE LENGTH OF A MINIMAL TEST SET

With the increasing size of hardware being built, it is not only important to minimize any particular circuit, but it is also vital to keep the cost of fault diagnosis as low as possible. In this chapter a lower and an upper bound on the length of a minimal test set for irredundant circuits will be derived. The problem of how to design circuits that require short testing procedures will also be discussed.

Let $\ell(T_s)$ denote the length of a minimal single fault test set T_s. For any irredundant network with p checkpoints T_s must detect 2p faults. Obviously, T_s cannot contain less than two tests, since no single test can detect both a s-0 and s-1 fault on the same line. On the other hand, if every fault should require a different test, $\ell(T_s)$ would be equal to 2p. These bounds, however, are quite rough and can be improved considerably. For networks consisting of AND, NAND, OR, NOR and NOT gates, the following two factors have great influence on the length of a minimal test set for fault detection.

1. All faults that belong to the same equivalence class are detected by the same test. Thus, one test is sufficient to detect s-0 faults on all inputs of an AND or NAND gate, and similarly only one test is sufficient to detect s-1 faults on all inputs of an OR or NOR gate.

2. Certain faults can never be detected by the same test. Faults of this type are nonequivalent faults on inputs of the same gate. For example, an AND gate with n inputs requires n different tests to detect the s-1 faults on all input leads. Similarly, an n-input OR gate needs n different input combinations to detect the s-0 faults. It should be pointed out, however, that two faults that belong to two different equivalence classes may be detected by the same test. For example, one test may detect s-1 faults on inputs of several AND gates in a two level AND-OR network.

The above discussion and most of the results presented in this chapter do not relate to networks built from other types of gates such as Exclusive-OR gates. It can be shown [16], that all single faults in any tree network consisting of only two input Exclusive-OR gates can be detected by four tests, where this number is independent of the size of the network. Unless otherwise mentioned, only networks that contain AND, NAND, OR, NOR and NOT gates will be dealt with.

Now consider irredundant fanout-free networks, or networks where fanout is present on the primary inputs only. Let N be such a network with p checkpoints. Any single fault test set for N must detect 2p faults. Denote these faults by $f_1, f_1', f_2, f_2', \ldots, f_p, f_p'$, where f_k is a s-0 fault and f_k' is a s-1 fault on checkpoint k, or vice versa.

<u>Lemma 6.1</u>: Let f_1 and f_2 be faults on any two checkpoints of a fanout-free network. If f_1 and f_2 are detected by the same test, then faults f_1' and f_2' cannot both be detected by the same test.

<u>Proof</u>: In a fanout-free network, faults are detected by checking for the presence of X_i or Y_j terms. Let fault f_1 cause the disappearance of one or more X_i terms. If f_2 can be detected by the same test as f_1, then there exists at least one term X_α such that X_α disappears due to f_1 as well as due to f_2. Because f_1 causes the disappearance of an X_i term, f_1' must make some Y_j term zero. Faults f_1' and f_2' can be both detected by the same test only if there exists at least one Y_j term that disappears due to both f_1' and f_2'. Faults f_1 and f_1', and faults f_2 and f_2', respectively, are faults on the same checkpoints. It follows from lemma 5.2 that f_1' and f_2' cannot be detected by the same test.

An <u>input limited fanout-free</u> (ILFF) network is defined to be a fanout-free network where the primary input fans out. Fanout-free networks where no primary input fans out are sometimes called <u>tree</u> networks. Bounds on the length of test sets for tree networks have been derived by Hayes [18a]. Theorem 6.1 specifies these bounds for ILFF networks.

<u>Theorem 6.1</u>: If T_s is a minimal single fault test set for an ILFF network with p checkpoints, where $p > 1$, then

$$2\sqrt{p} \leq \ell(T_s) \leq \frac{3}{2} p$$

<u>Proof</u>: The lower bound will be derived first. The test set T_s for a network with p checkpoints must detect 2p faults, denoted $f_1, f_1', f_2, f_2', \ldots, f_p, f_p'$. Suppose that r tests $t_1, t_2, \ldots, t_r$ are sufficient to detect faults $f_1, f_2, \ldots, f_p$. Let q_i be the number of faults detected by the test t_i, and t_i detect faults $f_1, f_2, \ldots, f_{qi}$, for $i = 1,2,\ldots,r$. It follows from lemma 6.1 that faults $f_1', f_2', \ldots, f_{qi}'$ must be detected by q_i different tests. In other words, $\ell(T_s)$ cannot be smaller than

$$r + \max(q_1, q_2, \ldots, q_r), \text{ where } \sum_{i=1}^{r} q_i \geq p$$

$\ell(T_s)$ will be minimal if each of the r tests detects the same number of faults, which is $\frac{p}{r}$. Hence the lower bound is given by the minimum of the function

$$h(r) = r + \frac{p}{r}$$

This function has a minimum for $r = \sqrt{p}$, since

$$h'(r) = 1 - pr^{-2}$$

$$h''(r) = 2pr^{-3}$$

Hence, $\ell(T_s) \geq \sqrt{p} + \frac{p}{\sqrt{p}} = 2\sqrt{p}$.

Now, the upper bound will be derived. T_s must detect 2p faults. However, (q+1) tests, rather than 2q tests, are sufficient to detect the 2q faults on checkpoints that are inputs to the same gate. Thus, at most r + p tests will be required, where r is the number of gates such that all inputs of the gate are checkpoints. Note that if all inputs of some gate are not checkpoints, then the equivalent faults on its inputs are detected by some test for the former. For any network with p checkpoints, where $p > 1$, r is at most $\frac{p}{2}$. Hence $\ell(T_s) \leq p + \frac{p}{2} = \frac{3}{2} p$.

The network in fig. 6.1 has a minimal test set that consists of eight tests. This is close to the upper bound, since $\frac{3}{2} p = \frac{3}{2} \cdot 6 = 9$. A minimal test set T_s for the network in fig. 6.2 contains four tests

$$T_s = (ab\bar{c}\bar{d},\ \bar{a}\bar{b}cd,\ \bar{a}b\bar{c}d,\ a\bar{b}c\bar{d})$$

The length of this test is identical with the lower bound, since $p = 4$ and $2\sqrt{p} = 4$.

The upper bound can further be improved. It is conjectured that the least upper bound is $p + \frac{p}{\log_2 p}$. However, the author has not been able to provide a rigorous proof (Appendix B). The results derived above are valid for fanout-free networks, or networks where only primary inputs fanout. The bounds for the general case of networks with internal fanout will now be derived.

<u>Corollary 6.1:</u> Let N be an irredundant single output network with p checkpoints that can be decomposed into k fanout-free subnetworks $N_1, N_2, \ldots, N_k$. Let p_i be the number

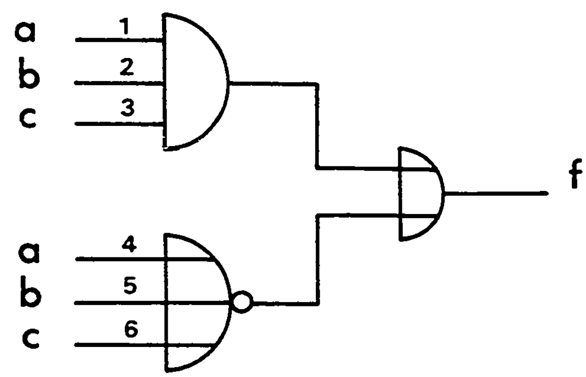


Figure 6.1 Network N6.1

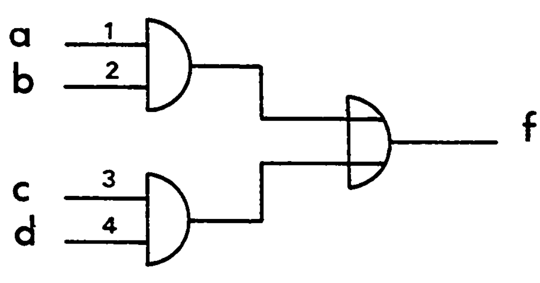


Figure 6.2 Network N6.2

of checkpoints in subnetwork N_i, for $i = 1,2,\ldots,k$, and let $p_{max} = \max (p_1,p_2,\ldots,p_k)$. The length of a minimal single fault test set T_s for network N satisfies the following inequality

$$2\sqrt{p_{max}} \le \ell(T_s) \le \frac{3}{2} p - 2(k-1)$$

Proof: If p_i is the number of checkpoints in subnetwork N_i, then by theorem 6.1 at least $2\sqrt{p_i}$ tests are needed to detect all single faults in N_i. Since $p_{max} = \max (p_1,p_2,\ldots,p_k)$, $\ell(T_s)$ must be greater than, or can at best be equal to, $2\sqrt{p_{max}}$. Similarly, at most $\frac{3}{2} p_i$ tests are required to detect all single faults in subnetwork N_i. If the output of N_i fans out to provide inputs for subnetwork N_j, then at most $\frac{3}{2}(p_i + p_j)-2$ tests are needed to detect faults in both N_i and N_j; note that at least two faults in N_j must be detected by tests for faults in N_i, otherwise it would not be possible to propagate the effect of faults in N_i through N_j. If N is a single output network that can be decomposed into k fanout-free subnetworks, then (k-1) subnetworks provide inputs for some other subnetwork. Hence $\ell(T_s)$ is at most $\frac{3}{2} p-2(k-1)$.

To illustrate, the bounds for the network in fig. 6.3 will be derived. This network has eight checkpoints and it can be decomposed into three fanout-free subnetworks, where $p_1 = 2$, $p_2 = 2$ and $p_3 = 4$. The lower bound is $2\sqrt{p_{max}} = 4$,

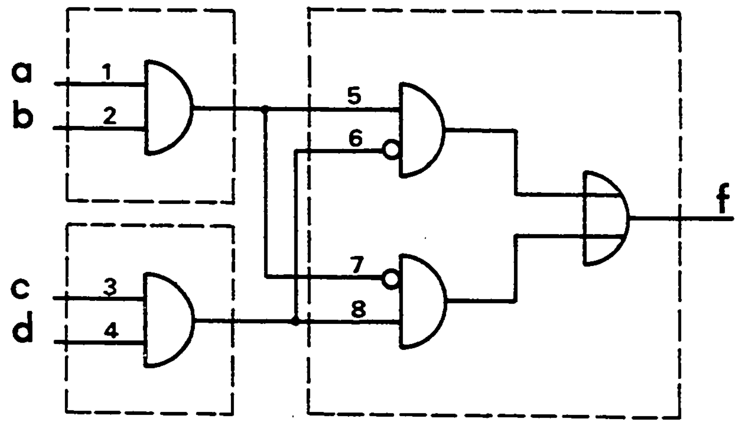


Figure 6.3 Network N6.3

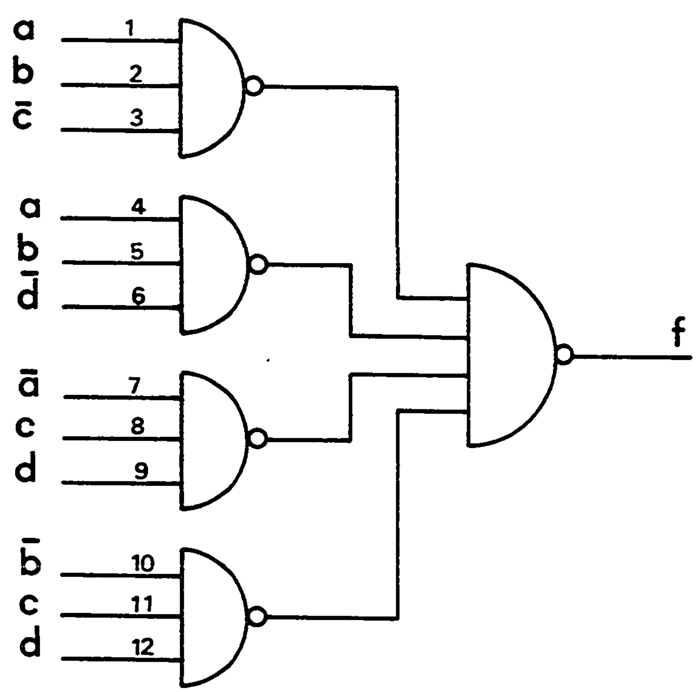


Figure 6.4 Network N6.4

the upper bound is $\frac{3}{2} \cdot 8 - 2 \cdot 2 = 8$. The network in fig. 6.4 has a minimal single fault test set of length four, and a minimal multiple fault test of length five, as was shown in chapters 2 and 3.

It should be pointed out that the lower bound is also a valid bound on the length of a minimal multiple fault test set T_m, since for any network $\ell(T_m) \geq \ell(T_s)$. It is more difficult to prove that $\ell(T_m)$ satisfies also the upper bounds derived above.

The structure of the network has a great influence on the length of a minimal test set. It was shown that the number of tests depends on the number of checkpoints in the network. Switching functions that are decomposable can be realized by a multiple level network with a smaller number of checkpoints than a two level realization corresponding to the minimal sum of products (or product of sums) form of the function. Furthermore, in such multiple level realizations a single test can be expected to detect a larger number of faults than a single test for the two level network. For example, the network in fig. 6.4 realizes the function $f = ab\bar{c} + ab\bar{d} + \bar{a}cd + \bar{b}cd$. This network has twelve checkpoints and needs at least nine tests to detect all faults. The output function, however, can be decomposed in the following way

$$f = ab(\bar{c} + \bar{d}) + cd(\bar{a} + \bar{b})$$

denoting $g = ab$, $h = cd$, f can be written as

$$f = g\bar{h} + \bar{g}h$$

The realization in fig. 6.4 is based on this decomposition. The circuit has eight checkpoints and it needs only four and five tests for single and multiple fault detection, respectively. This compares favourably with the nine tests required for the two level network. Similarly, a two level AND-OR network realizing the function

$$f = abcdeg + \bar{a}\bar{b}\bar{c}\bar{d}\bar{e}\bar{g} + abc\bar{d}\bar{e}\bar{g} + \bar{a}\bar{b}\bar{c}deg$$

has a minimal single fault test set of length 28. A three level network utilizing the following decomposition

$$f = (abc + \bar{a}\bar{b}\bar{c})(deg + \bar{d}\bar{e}\bar{g})$$

needs only 14 tests. Hence, from the testing point of view, multiple level design based on decomposition of switching functions is highly desirable. It should be noted, however, that all functions are not decomposable. The problem of decomposition of switching functions is presented elsewhere [3,23].

Chapter 7

FAULT DIAGNOSIS

When a fault is detected, it is necessary to identify the faulty component so that it can be replaced . Fault diagnosis will be considered under the single fault assumption. The problem of fault diagnosis (or fault location) is to generate a test set such that the location of the fault can be determined from the response of the network.

Because of the indistinguishability of certain faults, it is not always possible to uniquely specify which connection in the network is faulty. Two faults, f_1 and f_2, are said to be <u>distinguishable</u>, if there exists a test that detects f_1 and does not detect f_2, or vice versa. Schertz [44] has shown how to combine indistinguishable faults into equivalence classes. In a fanout-free network where inputs can be assigned independently while testing, two faults, f_1 and f_2, belong to the same equivalence class if and only if they are not distinguishable. In [44] only the structure of the network has been considered when the equivalence classes has been derived; when the values of input variables cannot be assigned independently, there may be indistinguishable faults that are not equivalent in the sense of [44]. For example, faults 1-1 and 4-1 in the network in fig. 7.1 do not belong to the same equivalence class. If the inputs are assigned as indicated in (a), then faults 1-1 and 4-1 are distinguishable; however if the inputs are assigned as in (b), then the two faults are not distinguishable. Since the input applied to checkpoint 1 is not independent of the input applied to checkpoint 3 (and input at 2 is not independent of the input at 4), the network in fig. 7.1 may not be considered as fanout-free. In the following section, it will be assumed that all inputs of a fanout-free network are accessible for testing.

7.1 Fanout-free Networks

It is difficult to derive a minimal diagnostic test set, unless all pairs of faults to be distinguished are considered. However, a nearly minimal diagnostic test set can be often derived with computational effort comparable to that required for fault detection.

Lemma 7.1: Let N be a fanout-free network such that both its ESP forms are irredundant. If a test set T checks for the presence of all X_i and all Y_j terms, then T distinguishes between any two single faults that do not belong to the same equivalence class.

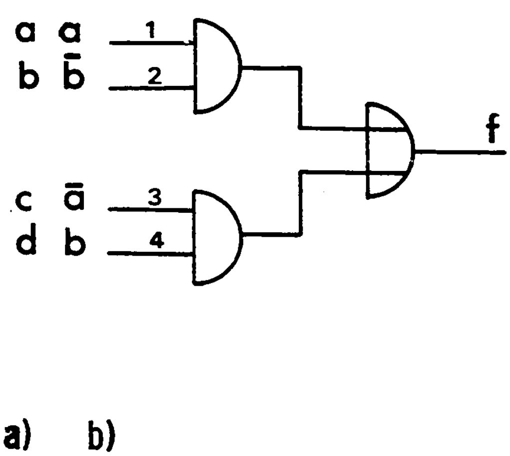


Figure 7.1 Network N7.1

Proof: If both the ESP(f) and ESP($\bar{f}$) forms are irredundant, then it is possible to check for the presence of all X_i and Y_j terms and T will detect all single faults in N. Clearly, any two faults f_1 and f_2, such that f_1 causes some X_i to disappear while f_2 makes some Y_j disappear, are distinguished, because such faults can never be detected by the same test. Now, suppose that fault f_1 causes the disappearance of terms $X_1, X_2, \ldots, X_p$ and that all these terms disappear due to fault f_2 as well. If no other X_i term disappears due to f_2, then f_1 and f_2 are equivalent because their effect upon the output function is the same. Otherwise there must be at least one X_i term that disappears due to f_2 and does not disappear due to f_1, and then T will contain at least one test that will distinguish f_1 from f_2.

A test set that checks for the presence of all X_i and Y_j terms is not, in general, a minimal diagnostic test set; however, an optimal test set for fault diagnosis is a subset of such a set. It is conjectured that in order to distinguish between any two distinguishable faults within a network, it is sufficient to distinguish between any two distinguishable faults on the checkpoints. A test that checks for the presence of some term detects faults on all checkpoints associated with that term. Therefore, the problem of generating an optimal diagnostic test set is equivalent to the problem of selecting a minimal set of

X_i and Y_j terms, such that for any two nonequivalent faults f_1 and f_2, this set contains at least one term which disappears due to f_1 and does not disappear due to f_2, or vice versa. This is illustrated by the following example.

Example 7.1: A diagnostic test set will be derived for the network in fig. 7.2, where the checkpoints are enumerated 1-7, while the remaining lines 8-12. The ESP forms are

$ESP(f) = \bar{a}_1\bar{b}_2\bar{c}_3 + \bar{a}_1\bar{b}_2\bar{d}_4 + e_5\bar{g}_6 + e_5\bar{h}_7$

$ESP(\bar{f}) = a_1\bar{e}_5 + a_1g_6h_7 + b_2\bar{e}_5 + b_2g_6h_7 + c_3d_4\bar{e}_5 + c_3d_4g_6h_7$

We denote by A_i the set of checkpoints associated with the term X_i.

Thus $A_1 = (1,2,3)$

A test set that checks for presence of $X_1 = \bar{a}_1\bar{b}_2\bar{c}_3$ detects faults 1-1,2-1 and 3-1. Because faults 1-1 and 2-1 are equivalent, we choose checkpoint 1 to represent this equivalence class. Then, the A sets are

$$A_1 = (1,3) \quad A_2 = (1,4) \quad A_3 = (5,6) \quad A_4 = (5,7)$$

Denoting by B_j the set of checkpoints associated with the term Y_j and choosing checkpoints 3 and 6 to represent the equivalence classes (3-0,4-0) and (6-0,7-0), respectively, the following B sets are obtained

$$B_1 = (1,5) \quad B_2 = (1,6) \quad B_3 = (2,5) \quad B_4 = (2,6)$$

$$B_5 = (3,5) \quad B_6 = (3,6)$$

It is now necessary to find a minimal set S_A of A sets such that for any two checkpoints p and q, where $p,q, \varepsilon\ (1,3,4,5,6,7)$,

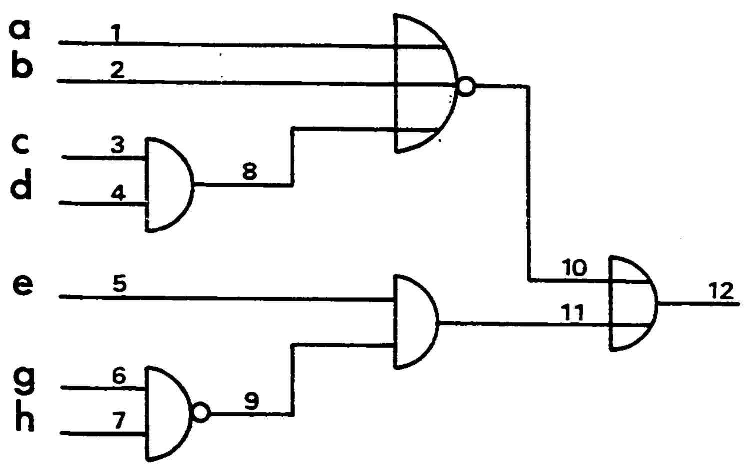


Figure 7.2 Network N7.2

TABLE 7.1

Fault Dictionary for Network N7.2

Fault in Equiv. Class	Failing Test(s)
1-0	t_5
2-0	t_6
(1-1,2-1,8-1,10-0)	t_1,t_2
3-1	t_1
4-1	t_2
(3-0,4-0,8-0)	t_7,t_8
5-1	t_5,t_6,t_7
(5-0,9-0,11-0)	t_3,t_4
6-1	t_3
7-1	t_4
(6-0,7-0,9-1)	t_8
(10-0,11-1,12-1)	t_5,t_6,t_7,t_8
12-0	t_1,t_2,t_3,t_4

S_A contains at least one A set that includes p but does not include q, or vice versa. It is easy to see that all four A sets must be chosen, i.e.

$$S_A = (A_1, A_2, A_3, A_4)$$

However, only four B sets are sufficient and the set S_B is

$$S_B = (B_1, B_3, B_5, B_6)$$

A test set T that checks for the presence of corresponding X_i and Y_j terms is

$$T = (\bar{a}\bar{b}\bar{c}d\bar{e}gh,\ \bar{a}\bar{b}c\bar{d}egh,\ abcde\bar{g}h,\ abcdeg\bar{h},\ a\bar{b}\bar{c}\bar{d}\bar{e}\bar{g}\bar{h},$$
$$\bar{a}b\bar{c}\bar{d}\bar{e}\bar{g}\bar{h},\ \bar{a}\bar{b}cd\bar{e}\bar{g}\bar{h},\ \bar{a}\bar{b}cdegh) =$$
$$= (t_1, t_2, \ldots, t_8)$$

Table 7.1 represents the fault dictionary for network N7.2 and the above test set.

In fanout-free networks where AND and OR gates alternate, every single fault equivalence class spans at most one gate. Thus, every fault can be identified to within a pair of interconnections in the network. A method that finds such fault-locatable realizations was described by Friedman and Menon [16].

7.2 Networks With Reconvergent Fanout

Faults in circuits with reconvergent fanout cannot be located as closely as in fanout-free networks. Even if the inputs can be assigned independently while testing, there may be indistinghishable faults in the same fanout-free subnetwork, as well as indistinguishable faults in two different fanout-free subnetworks. For example, in the network in fig. 7.3 faults 1-1 and 4-1 are indistinguishable and so are faults 2-0 (or 3-0 or 7-0) and 10-0.

Currently available methods for fault diagnosis [7,9] usually assume that a complete or a partial fault table is given, and they use this table to select the diagnostic test set. Thus, considerable effort is needed to build the fault table and additional effort is required to select the test set. A method for deriving an optimal test set for diagnosis without constructing the fault table of the circuit will now be described.

Let us denote by $f(f_a)$ the function realized by the circuit in the presence of the fault f_a. The set $D(f_a,f_b)$, of all tests that distinguish between faults f_a and f_b, is

$$D(f_a,f_b) = f(f_a) \oplus f(f_b) = \bar{f}(f_a) \oplus \bar{f}(f_b)$$

The function $f(f_a)$ can be easily obtained from the ESP(f) form of the network. The effect of faults upon the terms of the ESP forms has been established in chapter 2, and the following is a brief summary of how a s-0 or s-1 fault on checkpoint p of a normal network influences term $a_{pq...s}W$:

1. In the presence of the fault p-1

   term $a_{pq...s}W$ changes to $1.W = W$

   term $\bar{a}_{pq...s}W$ changes to $\bar{1}.W = 0.W = 0$

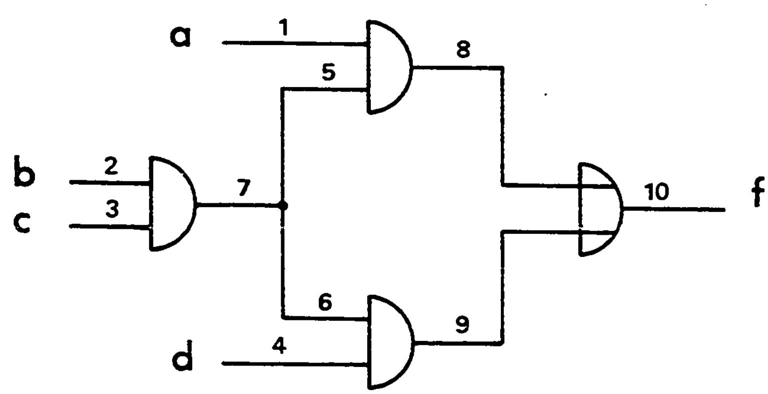


Figure 7.3 Network N7.3

2. In the presence of the fault p-0

term $\bar{a}_{pq...s}W$ changes to $\bar{0}.W = 1.W = W$

term $a_{pq...s}W$ changes to $0.W = 0$

Similarly, faults q-1 and q-0 have the same effect as faults p-1 and p-0, respectively, provided that the inversion parity between p and q is even, and the opposite effect when the inversion parity is odd. Thus, the function f(p-1) or f(p-0) is obtained from the ESP(f) form by applying the rules above to every literal that is associated with checkpoint p, removing all subscribts, and using all the Boolean identities to simplify the resulting expression.

To generate a diagnostic test set by considering all pairs of faults on checkpoints would require a large amount of computation. Because a complete test set for fault diagnosis has to detect all faults, it is reasonable to expect that a test set for fault detection is a subset of an optimal diagnostic test set. Assuming that a single or multiple fault test T_1 has been derived, the following procedure generates a test set T_2, such that $(T_1 \cup T_2)$ is an optimal test set for fault location.

Procedure 7.1:

1. From the checkpoint covering table, find the set of all fault pairs (f_a, f_b), such that faults f_a and f_b are not distinguished by T_1. When forming these pairs, select one fault to represent each class of equivalent faults.

2. Use ESP(f) or ESP($\bar{f}$), depending on which has a smaller number of terms, to evaluate $f(f_a)$ or $\bar{f}(f_a)$ in the manner described above (step 3 assumes ESP(f) used).

3. For every pair (f_a, f_b) compute

$$D(f_a, f_b) = f(f_a) \oplus f(f_b)$$

4. Test set T_2 is any covering, preferably minimal, of all the $D(f_a, f_b)$ terms computed in step 3.

It should be noted that if $D(f_a, f_b) = \emptyset$ for some f_a and f_b, then faults f_a and f_b are not distinguishable.

TABLE 7.2

Test Set T_1 for Network N7.4

	1	2	3	4	5	6	7	8
$a\bar{b}\bar{c}\bar{d}$	0	-	1	-	-	-	-	1
$a\bar{b}cd$	0	0	-	-	1	-	-	1
$abcd$	0	0	-	-	0	0	1	-
$abc\bar{d}$	-	-	-	-	-	1	-	1
$a\bar{b}\bar{c}d$	-	1	0	1	-	-	-	0
$\bar{a}bcd$	1	-	-	0	-	-	0	-

Example 7.2: Test set T_1 (see Table 7.2) is a minimal test set which detects all faults in the network N7.4. Choosing faults 3-0,4-0, and 5-0 to represent the equivalence classes (3-0,2-1), (4-0,7-0) and (5-0,6-0), respectively, it can be observed from table 7.2 that the following fault pairs are not distinguished by T_1: (3-0,8-0), (1-1,4-0),

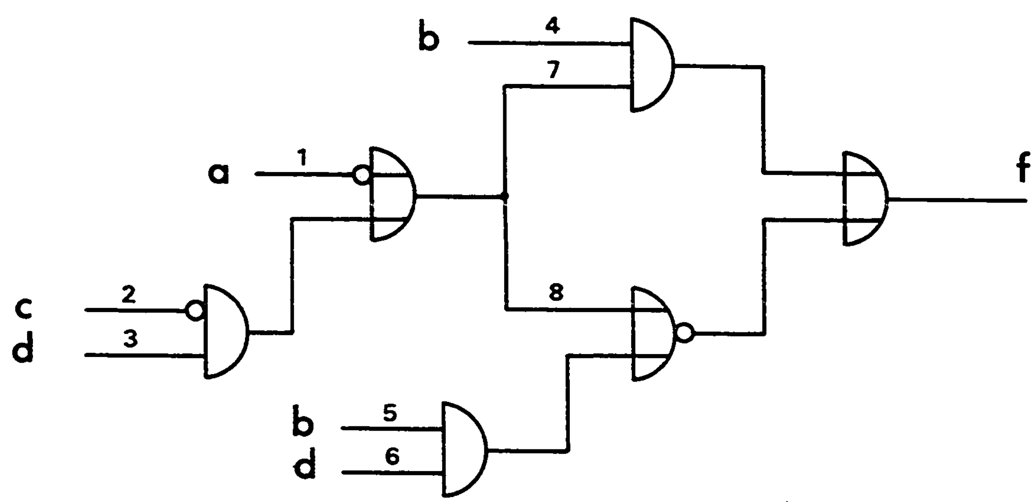


Figure 7.4 Network N7.4

(4-1,3-0), (4-1,8-0) and (7-1,5-0). The ESP(f) form of this network is

$$ESP(f) = \bar{a}_{17}b_4 + b_4\bar{c}_{27}d_{37} + a_{18}\bar{b}_5c_{28} + a_{18}\bar{b}_5\bar{d}_{38} + a_{18}c_{28}\bar{d}_6 + \\ + a_{18}\bar{d}_{38}\bar{d}_6$$

For the pair (3-0,8-0)

$$f(3\text{-}0) = \bar{a}b + 0 + a\bar{b}c + a\bar{b} + ac\bar{d} + a\bar{d} =$$

$$= \bar{a}b + a\bar{b} + a\bar{d}$$

When evaluating f(8-0), note that the inversion parity between 3 and 8 is even, whereas between 2 and 8, or 1 and 8, it is odd.

$$f(8\text{-}0) = \bar{a}b + b\bar{c}d + \bar{b} + \bar{b} + \bar{d} + \bar{d} =$$

$$= \bar{a} + \bar{b} + \bar{d} + \bar{c}$$

$$D(3\text{-}0,8\text{-}0) = (\bar{a}b + a\bar{b} + a\bar{d}) \oplus (\bar{a} + \bar{b} + \bar{d} + \bar{c}) =$$

$$= \bar{a}\bar{b} + ab\bar{c}d$$

In the same way it is found that

$$D(1\text{-}1,4\text{-}0) = \bar{a}\bar{d} + b\bar{c}d + \bar{a}\bar{b}c$$

$$D(4\text{-}1,3\text{-}0) = \bar{a}\bar{b} + ab\bar{c}d + abc\bar{d}$$

$$D(4\text{-}1,8\text{-}0) = \emptyset$$

$$D(7\text{-}1,5\text{-}0) = \emptyset$$

Hence the fault 4-1 is not distinguishable from fault 8-0, and 7-1 is not distinguishable from 5-0. Each of the tests $ab\bar{c}d$, $\bar{a}\bar{b}c$, or $\bar{a}\bar{b}\bar{d}$ covers all the nonzero terms. Choosing $T_2 = ab\bar{c}d$, the resulting test set is

$$T_1 \cup T_2 = (a\bar{b}\bar{c}\bar{d},\ a\bar{b}cd,\ abcd,\ abc\bar{d},\ a\bar{b}\bar{c}d,\ \bar{a}bcd,\ ab\bar{c}d) =$$

$$= (t_1, t_2, t_3, \ldots, t_7)$$

TABLE 7.3

Fault Dictionary for Network N7.4

Fault in Equiv. Class	Failing Test(s)
1-1	t_6
2-0	t_2,t_3
3-1	t_1
(2-1,3-0,9-0)	t_5,t_7
(1-0,9-1,10-1)	t_1,t_2,t_3
10-0	t_5,t_6,t_7
4-1	t_5
7-1	t_3
(4-0,7-0,12-0)	t_6,t_7
5-1	t_2
6-1	t_4
(5-0,6-0,11-0)	t_3
8-0	t_5
(8-1,11-1,13-0)	t_1,t_2,t_4
(12-1,13-1,14-1)	t_3,t_5
14-0	t_1,t_2,t_4,t_6,t_7

Test $ab\bar{c}d$ checks for the presence of the term $X_2 = b_4\bar{c}_{27}d_{37}$. This term disappears due to the following single faults: 4-0,2-1,3-0 and 7-0. Network N7.4 is not a network with EIP, but since no X_i term can grow due to fault 2-1 or 3-1 into X_i' such that $ab\bar{c}d \in X_i'$, test $ab\bar{c}d$ detects all the single faults above. From this fact and from Table 7.2, the fault dictionary in Table 7.3 is constructed.

With modern LSI technology the degree of diagnostic resolution required is usually determined by the smallest replacable modules of the system. Hence, there is no need to distinguish any two faults that belong to the same module and the diagnostic test set would be generated by considering only those fault pairs (f_a,f_b), where fault f_a and fault f_b are faults in two different modules.

Chapter 8

CONCLUSIONS

The methods developed in this thesis derive the test set by checking for growth and/or presence of the terms in the ESP form(s) of the network. The ESP form describes the function realised by the network while preserving the important structural properties. It is for these reasons the ESP form is appropriate for test generation. A brief summary of the advantages of the approach taken herein is now presented, together with a discussion of the disadvantages, and suggestions for ways of circumventing them. It appears that the main advantages are as follows:

1. Single fault detection. The test set generated by procedure 2.1 or 5.1 is minimal. The computational effort required by procedure 5.1 compares very favourably with almost any other method.

2. Multiple fault detection. It is felt that the ESP form is able to solve the multiple fault problem more easily than the other methods, such as those based on path sensitizing. The results in section 3.1 provide an insight into conditions under which fault masking occurs and the

results are also used to obtain minimal multiple fault test set (procedure 3.3). For most practical purposes it may be preferable to derive a nearly minimal test by utilizing the result of theorem 3.3, or theorem 3.4, since the computational effort is considerably smaller.

When a single fault test set is derived for networks with unequal inversion parity of all reconverging paths, both the ESP(f) and ESP($\bar{f}$) forms are employed by procedure 2.1. However, the number of terms in at least one of the two forms may be large for some networks. A test set for such networks can be also derived by checking for the presence and growth of only X_i (or only Y_j) terms, provided that it is ensured that reconvergent fanout cancellation does not occur.

If the ESP form is redundant, it may not be possible to check for presence of every term. If this is the case, it is suggested to extend procedure 5.1 in the following way (assume ESP(f) is used):

- generate tests that check for growth of all X_i terms, and tests that check for presence of every term X_α such that $X_\alpha \cdot \prod_{i \neq \alpha} \bar{X}_i \neq \emptyset$.

- if a fault(s) is not detected by the tests generated above, find a pair, or triple, etc..., of terms that all disappear due to this fault and generate a test that checks for presence of such a pair, or triple, etc., of terms.

Theorem 3.3 is very useful when only one of the two ESP forms is used and a multiple fault test set is to be derived. However, this theorem cannot be applied when it is not possible to strongly check for growth of all terms. A valuable extension of this work would be to prove that the theorem also holds for the more general case of the weak check.

The problem of checking experiments for sequential machines has not been considered in this thesis. Although a number of methods for generating such testing sequences have been described [16,19,22,26,46], no satisfactory solutions are available. This topic is also worthy of further research.

REFERENCES

REFERENCES

[1] Amar, V. and Condulmari, N., "Diagnosis of Large Combinational Networks", IEEE Trans. on Elec. Computers, vol. EC-16, pp. 675-680, October 1967.

[2] Armstrong, D. B., "On Finding a Nearly Minimal Set of Fault Detection Test for Combinational Logic Nets", IEEE Trans. on Elec. Computers, vol. EC-15, pp. 66-73, February 1966.

[3] Ashenhurst, R. L., "The Decomposition of Switching Functions", Proceedings of the Intern. Symposium on the Theory of Switching, April 2-5, 1957.

[4] Bearnson, L. W. and Carrol, C. C., "On the Design of Minimum Length Fault Test for Combinational Circuits", IEEE Trans. on Computers, vol. C-20, pp. 1353-1356, November 1971.

[5] Bossen, D. C. and Hong, S. J., "Cause-Effect Analysis for Multiple Fault Detection in Combinational Networks", IEEE Trans. on Computers, vol. C-20, pp. 1252-1257, November 1971.

[6] Brule, J. D., Johnson, R. A., and Kletsky, E., "Diagnosis of Equipment Failures", IRE Trans. on Reliability and Quality Control, vol. RAC-9, pp. 23-24, April 1960.

[7] Chang, H. Y., "An Algorithm for Selecting an Optimum Set of Diagnostic Tests", IEEE Trans. on Elec. Computers, vol. EC-14, pp. 706-710, October 1965.

[8] Chang, H. Y., "A Distinguishability Criterion for Selecting Efficient Diagnostic Tests", AFIPS Proc. of Spring Joint Comp. Conf., vol. 32, pp. 529-534, 1968.

[9] Chang, H. Y., Manning, E. G. and Metze, G., Fault Diagnosis in Digital Systems, Wiley-Interscience, 1970.

[10] Chiang, A.C.L., Reed, J. S. and Barnes, A. V., "Path Sensitization, Partial Boolean Difference and Automated Fault Diagnosis", IEEE Trans. on Computers, vol. C-21, pp. 185-195, February 1972.

[11] Curtis, H. A., A New Approach to The Design of Switching Circuits, D. Van Nostrand Company, Inc., Princeton, N. J., 1962.

[12] Du, Min-Wen and Weiss, C. D., "Multiple Fault Detection in Combinational Circuits:Algorithms and Experimental Results", Digest of Papers, 1972 Int. Symp. on Fault-Tolerant Computing, pp. 120-125, Newton, Mass., June 19-21, 1972.

[13] Eldred, R. D., "Test Routines Based on Symbolic Logic Statements", Journ. of ACM, Vol. 6, pp. 33-36. January 1959.

[14] Fridrich, M. and Davis, W. A., "Single Fault Detection in Combinational Circuits", Session 72, Conference Record of The Canadian Computer Conference, Montreal, June 1-3, 1972.

[15] Friedman, A. D., "Fault Detection in Redundant Circuits", IEEE Trans. on Elec. Comp., vol. EC-16, pp. 99-100, February 1967.

[16] Friedman, A. D. and Mennon, R., Fault Detection in Digital Circuits, Prentice-Hall, Inc., M.J., 1971.

[17] Gault J. W., Robinson J. P. and Reddy, M. S., "Multiple Fault Detection in Combinational Networks," IEEE Trans. on Computers, vol. C-21, pp. 31-36, January 1972.

[18] Hayes, J. P., "A NAND Model for Fault Diagnosis in Combinational Logic Networks", IEEE Trans. on Computers, vol. C-20, pp. 1496-1505, December 1971.

[18a] Hayes, J. P., "On Realizations of Boolean Functions Requiring a Minimal or Near-Minimal Number of Tests," IEEE Trans. on Computers, vol. C-20, pp. 1506-1513, December 1971.

[19] Hennie, F. C., "Fault Detecting Experiments for Sequential Circuits", Proc. of the 5th Ann. Symp. on Switching Theory and Log. Design, pp. 95-110, October, 1964.

[20 Hornbuckle, G. D. and Spann, R. N., "Diagnosis of Single-Gate Failures in Combinational Circuits", IEEE Trans. on Computers, vol. C-18, pp. 216-220, March 1969.

[21] Howarter, D. R., "The Selection of Failure Location Tests by Path Sensitising Techniques", CSL Report R-316, Univ. of Illinois, Urbana, August 1966.

[22] Hsieh, E. P., "Checking Experiments for Sequential Machines", IEEE Trans. on Computers, vol. C-20, pp. 1152-1166, October 1971.

[23] Karp, R. M., "Functional Decomposition and Switching Circuit Design", SIAM Journ., pp. 291-335, June 1963.

[24] Kautz, W. H., "Testing for Faults in Cellular Logic Arrays", Proc. 8th Ann. Symp. on Switching and Aut. Theory, pp. 161-174, 1967.

[25] Kautz, W. H., "Fault Testing and Diagnosis in Combinational Logical Circuits", IEEE Trans. on Computers, vol. C-17, pp. 352-367, April 1968.

[26] Kohavi, Z. and Lavellee, P., "Design of Sequential Machines with Fault Detection Capabilities", IEEE Trans. on Elec. Comp., vol. EC-16, pp. 473-484, August 1967.

[27] Kohavi, I. and Kohavi, Z., "Variable-Length Distinguishing Sequences and Their Application to the Design of Fault-Detection Experiments", IEEE Trans. on Computers, vol. C-17, pp. 792-795, August 1968.

[28] Kohavi, I., "Fault Diagnosis of Logical Circuits", IEEE Conf. Rec. of 1969 Tenth Ann. Symp. on Switching and Aut. Theory, pp. 166-173, October 1969.

[29] Kohavi, I. and Kohavi, Z., "Detection of Multiple Faults in Combinational Logic Networks", IEEE Trans. on Computers, vol. C-21, pp. 556-567, June 1972.

[30] Manning, E. G., "Self-Diagnosis of Electronic Computers - An Experimental Study", CSL Report R-259, Univ. of Illinois, Urbana, May 1969.

[31] Marinos, P. N., "Derivation of Minimal Complete Sets of Test-Input Sequences Using Boolean Differences", IEEE Trans. on Computers, vol. C-20, pp. 25-32, January 1971.

[32] Millman, J. and Taub, H., Pulse, Digital and Switching Waveforms", Mac-Graw Hill, Inc., N.Y., 1965.

[33] Paige, M. R., "Generation of Diagnostic Tests Using Prime Implicants", CSL Report R-414, Univ. of Illinois, Urbana, May 1969.

[34] Peterson, W. W., Error-Correcting Codes, MIT Press and John Wiley and Sons, Inc., N.Y., 1962.

[35] Poage, J. F., "Derivation of Optimum Tests to Detect Faults in Combinational Circuits", Proc. of the Symp. of Math. Theory of Automata, pp. 483-528, April 24-26, 1962.

[36] Poage, J. F. and McCluskey, E. Y., "Derivation of Optimum Tests Sequences for Sequential Machines", Proc. of the 5th Ann. Symp. on Switching Theory and Logical Design, Princeton University, pp. 121-132, Nov. 11-13, 1969.

[37] Powell, T. J., "A Procedure for Ranking Diagnostic Test Inputs", CSL Report R-354, Univ. of Illinois, Urbana, May 1967.

[38] Powell, T. J., "A Module Diagnostic Procedure for Combinational Logic", CSL Report R-413, Univ. of Illinois, Urbana, April 1969.

[39] Preparata, F., Metze, G. and Chien, R., "On the Connection Assignment Problem of Diagnosable Systems", IEEE Trans. on Elec. Comp., vol. EC-16, pp. 848-854, December 1957.

[40] Ramamoorthy, C. V., "A Structural Theory of Machine Diagnosis", Proc. of Spring Joint Comp. Conf., vol. 30, pp. 743-756, 1967.

[41] Roth, J. P., "Diagnosis of Automata Failures: A Calculus and a Method", IBM Journ. of Research and Development, vol. 10, pp. 279-291, July 1966.

[42] Roth, J. P., Bonricius, W. G. and Schneider, P. R., "Programmed Algorithms to Compute Tests to Detect and Distinguish Between Failures in Logic Circuits", IEEE Trans. on Elec. Comp., vol. EC-16, pp. 567-579, October 1967.

[43] Schertz, D. R. and Metze, G. A., "On the Indistinguisability of Faults in Digital Systems", Proc. of the 6th Allerton Conf. on Circuit and System Theory, pp. 752-760, October 1968.

[44] Schertz, D. R., "On the Representation of Digital Faults", CSL Report R-418, Univ. of Illinois, Urbana, May 1969.

[45] Sellers,F. F., Hsiao, M. Y. and Bearnson, L. W., "Analysing Errors with the Boolean Difference", IEEE Trans. on Computers, vol. C-17, pp. 676-683, July 1968.

[46] Seshu, S. and Freeman, D. N., "The Diagnosis of Asynchronous Sequential Switching Systems", IRE Trans. on Elec. Comp., vol. EC-11, pp. 459-465, August 1962.

[47] Seshu, S., "On an Improved Diagnosis Program", IEEE Trans. on Elec. Comp., vol. EC-14, pp. 76-79, January 1965.

[48] Yau, S. S. and Tang, Y. S., "An Efficient Algorithm for Generating Complete Test Sets for Combinational Logic Circuits", IEEE Trans. on Computers, vol. C-20, pp. 1241-1251, November 1971.

APPENDIX

APPENDIX A

VALIDITY OF PROCEDURE 3.1

Procedure 3.1 finds the multiple faults that are not detected by a single fault test set T_s by utilizing the result of Theorem 3.2. It was shown in the proof of this theorem, that if a test t detects fault f_1 and this fault is masked, then there exists a term Y_α (or X_β) which grows into Y'_α (or X'_β) and $t \in Y'_\alpha$ (or $t \in X'_\beta$). Procedure 3.1 constructs the intersection $P = X'_i.\bar{X}_i.f.T_s$ (or $Y'_j.\bar{Y}_j.f.T_s$), although it has not been proved that $t \in Y'_\alpha.\bar{Y}_\alpha$ (or $t \in X'_\beta.\bar{X}_\beta$).

In a fanout-free network every variable is associated with only one checkpoint. Therefore, if a term disappears due to fault f_1, it cannot grow due to fault f_2 when f_1 is present. Consequently, $t \in Y'_\alpha.\bar{Y}_\alpha$ (or $t \in X'_\beta.\bar{X}_\beta$) holds for fanout-free networks.

For networks with internal fanout a term which disappears due to fault f_1 may grow due to fault f_2 when (f_1,f_2) occurs. If this is the case, then f_1 and f_2 are faults in two different fanout-free subnetworks, say subnetwork N_1 and N_2, such that the output of N_1 fans out to provide inputs for N_2. Suppose that fault f_1 makes the term $Y_\alpha = UW$ disappear by making some literal in U, become zero and that f_1 is detected by test $t_1 \in Y_\alpha$. If Y_α grows into $Y'_\alpha = W$ due to fault f_2, then f_2 masks f_1 under t_1, but $t_1 \notin Y'_\alpha.\bar{Y}_\alpha$. We shall now show that procedure 3.1 does not miss any undetected multiple faults for the following reasons:

1. Fault f_1 is not equivalent to a fault on the output of subnetwork N_1. Then there will be at least one additional Y_j term, say term Y_γ, where $Y_\gamma = VW$ and $U.V = \emptyset$, and such that Y_γ grows due to f_2 into $Y'_\gamma = W$. If Y_α is not checked for growth due to f_2 by T_s, then Y_γ is not either, and procedure 3.1 will not miss this masking relation since $t_1 \in Y'_\gamma.\bar{Y}_\gamma$.

2. Fault f_1 is equivalent to a fault on the output of subnetwork N_1. Hence it is equivalent to a multiple fault $(f_3,f_4,\ldots,f_n)$ on inputs of N_2. Two possibilities can occur:

(a) Faults $f_3,f_4,\ldots,f_n$ do not make term Y_α disappear. Then test set T_s will contain at least one additional test that detects faults $f_3,f_4,\ldots,f_n$, and also f_1; hence f_2 will not mask f_1 under T_s.

(b) Faults $f_3,f_4,\ldots,f_n$ are detected by the same test as f_1, i.e. by t_1. Then they cause term $Y_\alpha = UW$ to disappear by making some literal in W zero and Y_α cannot grow due to f_2. Therefore f_1 will not be masked.

Thus, it is sufficient to construct the intersection $P = X'_i.\bar{X}_i.f.T_s$ (or $Y'_j.\bar{Y}_j.\bar{f}.T_s$) in order to find the multiple faults that are not detected by T_s.

APPENDIX B

UPPER BOUND ON $\ell(T_s)$ FOR ILFF NETWORKS

A single fault test set T_s for an ILFF network with p checkpoints must detect 2p faults, denoted f_1, $f_1, f_1', f_2, f_2', \ldots f_p, f_p'$. Let r tests detect faults $f_1, f_2, \ldots, f_p$ by checking for the presence of r X_i terms $X_1, X_2, \ldots, X_r$. Then at most p tests will be needed to detect faults $f_1', f_2', \ldots, f_p'$ and this will happen when all terms are a distance 3 apart. The distance between two terms X_1 and X_2 is the number of literals that appear in X_1 as well as X_2, such that the appearance is complemented in X_1 and not complemented in X_2, or vice versa. If n is the number of different literals in the ESP(f) form, then [34]

$$r \leq 2^{n-\log_2 n+1} \qquad \text{(b1)}$$

It is now conjectured that

$$r.\ n \leq p$$

if all the terms are to be a distance 3 apart.

Then $n \le \frac{p}{r}$ and from (b1)

$$r < 2^{\frac{p}{r}} - \log_2 \frac{p}{r}$$

$$\log_2 r < \frac{p}{r} - \log_2 p + \log_2 r$$

$$r < \frac{p}{\log_2 p}$$

Hence $\ell(T_s) \le r + p < p + \frac{p}{\log_2 p}$.