7918735

AJABNOOR, YOUSEF MOHAMMED
ANALYSIS OF MULTI-VALUED COMBINATIONAL
CIRCUITS USING DIGITAL CALCULUS.

THE UNIVERSITY OF OKLAHOMA, PH.D., 1979

University Microfilms International 300 N. ZEEB ROAD, ANN ARBOR, MI 48106

THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

ANALYSIS OF MULTI-VALUED COMBINATIONAL CIRCUITS USING DIGITAL CALCULUS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

degree of

DOCTOR OF PHILOSOPHY

By

YOUSEF M. AJABNOOR

Norman, Oklahoma

1979

ANALYSIS OF MULTI-VALUED COMBINATIONAL

CIRCUITS USING DIGITAL CALCULUS

APPROVED BY

DISSERTATION COMMITTEE

DEDICATION

To my parents, my wife Amal, and my daughter Maram

ACKNOWLEDGMENTS

Having completed this work, I would like to express my gratitude and extreme appreciation to my advisor Professor Samuel C. Lee for suggesting the topic of this dissertation and his continuous and knee supervision. I would also like to thank my committee members Professor M.Y. El-Ibiary, Professor J.A. Payne, Professor S.B. Eliason, and Professor B.A. Magurn for their comments and suggestions during the preparation of this dissertation.

ABSTRACT

In this dissertation, Boolean differential calculus is extended to a more general calculus which will be applicable to not only binary switching systems and Boolean switching systems, but also multi-valued switching systems. This generalized Boolean differential calculus is called multi-valued digital calculus, or simply digital calculus. In this generalization, the Boolean partial derivative and the Boolean total differential of a Boolean function are extended to multi-valued partial derivative and multi-valued total differential of a multi-valued function defined on a potentially implementable multi-valued algebra, respectively. A computer method for computing multi-valued partial derivative and multi-valued total differential is presented.

The main objective of this research is to investigate the theory that is necessary for developing non-binary digital systems. In particular, it is concerned with the development of the theory of digital calculus defined on potentially implementable multi-valued algebra and its applications to the design of hazard-free multi-valued digital systems.

Potentially implementable multi-valued algebras are discussed and the multi-valued digital systems derived from them are studied. In the consideration of multi-valued digital system design, multi-valued digital calculus is applied to the derivation

of a complete set of tests for detecting permanent faults on primary input and internal lines of a multi-valued combinational circuit of these systems. Examples illustrating fault detection using multi-valued digital calculus are given.

Two general procedures, one for static-hazard detection and one for static hazard-elimination after they are detected, in a multi-valued combinational circuit are presented. The detection procedure utilizes the equivalent normal form (ENF) of the given circuit and multi-valued digital calculus. The elimination procedure uses a technique of adding redundant circuitry to the given circuit. Examples illustrating these procedures are also given.

TABLE OF CONTENTS

Page

LIST OF TABLES

TABLE Page

LIST OF ILLUSTRATIONS

FIGURE Page

CHAPTER I

INTRODUCTION

1.1 Motivation

Today, we are living in a world of binary digital systems, such as digital watches, digital calculators, digital computers,...,etc. Especially, in the last few years, the technology of two-valued switching components which make up the binary digital systems has been greatly improved which makes binary digital systems cheap and reliable. However, the binary digital system is only a subset of a more general class of digital systems, namely multi-valued digital systems. Considerable advantages may be gained by considering multi-valued digital systems over binary digital systems (two-valued digital systems). For instance,

(1) The size and speed of binary circuit components and memory devices approaching to its limit. The binary number system has been used throughout the entire development of computer technology. The growth of computation systems and the need to process increasing volumes

of data faster has resulted in the development of large scale integration (LSI) integrated circuits. However, the volume of data continues to increase while the circuit components and memory devices approach to their practical limit in size and speed. The design of computation systems in other number systems seems to be a logical solution to the continued increase in volume of data to be processed by digital computers.

(2) Pin limitation problem of binary LSI.

One of the most important problems in designing very large binary LSI is the pin limitation of the integrated circuits. Multi-valued switching circuits allow each input pin to accept and each output pin to deliver more information; therefore, for the same amount of information transfer, the total number of pins required in an integrated circuit chip containing multi-valued switching elements is much less than that of an integrated circuit chip with binary elements. As a simple example, in binary system, eight input pins of an integrated circuit chip are sufficient to accept any of the 256 numbers whereas only six input pins are needed (a one-fourth pin number reduction) ternary (3-valued) switching circuit is used.

(3) The need of decimal-to-binary and binary-to-decimal conversions. Since man works with decimal numbers, it is desirable to have a computer which accepts decimal numbers, processes them, and produces decimal numbers directly to eliminate the process of converting back and fourth between decimal and binary numbers which are required in binary-numbers based computers.

(4) Increasing reliability and reducing cost of analog and digital circuitry. In the past, few multi-valued switching circuits have been built because of high cost and low reliability. However, the price for binary digital circuits has been decreasing and will continue to drop in the future. At the same time, the industry is building more reliable and testable multi-valued switching circuits.

Even though we all know that there are so many obvious advantages of using multi-valued digital systems over the binary digital systems such as those described above, the theory needed for developing such systems has not yet been developed. This research is devoted to the investigation of multi-valued switching system theory and its application to the design of reliable multi-valued digital systems.

1.2 Relevant Works and Statement of the Problems

No one will dispute that the biggest advance in the history of mathematics, which set the foundation for the beginning of the modern mathematics era, was the discovery of calculus by Newton and Leibniz. The discovery of ordinary calculus solved many problems in mathematics and related areas. Computer scientists have found that the development of Boolean differential calculus can solve many digital design problems in a systematic way and it makes these problems quite simple to solve. The concept of Boolean derivative (Boolean difference) of a Boolean function, introduced first by Reed [1] in 1954, has been more thoroughly investigated in two papers due to Akers [2] and to Sellers, Hsiao, and Bearson [3]. In both these papers, various differential operators are introduced and described in connection with their application to switching problems. Boolean differential calculus, introduced by

Thayse in 1971 [4] encompasses and generalizes the algebraic concepts introduced by the former researchers. Recently, Lee [5,6, Reference 7, Chapter 2 and 3] further extended it to binary-vector Boolean differential calculus. It has been shown [6] that any Boolean function can be analytically represented by a binary-vector switching function. Properties and canonical forms of it were presented.

The development of multi-valued digital systems began in 1920 with the work of Post [8] and Luikasiewiez [9]. Multi-valued switching algebra, introduced first by Rosenbloom [10] in 1942 and called Post Algebra of order m where $m \geq 2$, has been formulated in two papers due to Vranesic, Lee, and Smith [11] in 1970 and to Su and Sarries [12] in 1972. Multi-valued digital systems have been also attacked by many researchers [13-17].

The area of multi-valued switching systems is widely open for research; many problems need to be solved by the digital designers and researchers in order to make the multi-valued digital system practical. In this research, it is intended to find a mathematical foundation for the analysis and design of multi-valued digital systems by extending Boolean differential calculus into a more general calculus which will be applicable not only to binary switching systems, but also to multi-valued switching systems. In particular multi-valued digital calculus will be applied to solve fault detection, static hazard detection, and static hazard elimination problems in multi-valued combinational switching circuits.

1.3 Research Objectives and Description of the Research

The first objective of this research is to extend two-valued partial derivative and two-valued total differential into multi-valued partial derivative and multi-valued total differential, respectively, which includes a study of multi-valued switching functions that are completely characterized by the algebraic properties of their associated differentials. The second objective is to develop the theory of the multi-valued partial derivative and the total differential and their applications to the test generation for fault detection, static hazard detection, and static hazard elimination in multi-valued combinational switching circuits. The research can be described by the following three parts.

(1) Development of a method for deriving tests for detecting single faults in multi-valued combinational switching circuits. The fault testing and diagnosis of multi-valued circuits becomes an important part of the system design because the multi-valued switching circuits may be disabled by an error. The process of applying

tests and determining whether the circuit is fault free or not is generally known as fault-detection. Both single faults on input lines (primary lines) and on internal lines of a multi-valued combinational switching circuit are considered. Single faults detection in a multi-valued combinational switching circuit will be the first task of this research.

(2) Development of a method for detecting static hazards in asynchronous multi-valued combinational switching circuit whenever the input signals of a multi-valued combinational switching circuit are changed, (i.e., the transient conditions), the use of the truth table associated with the circuit can lead to an incorrect behaviour of the network, this incorrect behaviour is called hazard. Hazards may be static hazards or dynamic hazards, but only static hazards is considered in this research. Moreover, static hazard detection is not only considered for two-level combinational circuits which is considerably simple, but it is also considered for multi-level multi-valued combinational switching circuits. The proposed technique

is to develop an accurate mathematical model that is based on a systematic use of multi-valued partial derivative and multi-valued total differential to detect static hazard in multi-valued combinational switching circuits.

(3) Developing the theory for eliminating static hazards in asynchronous multi-level multi-valued combinational switching circuits. From a practical point of view, it is not enough just to know the existance of the static hazard in a circuit without removing it. Thus, the third task of this research is to eliminate all static hazards present in asynchronous multi-level multi-valued combinational switching circuits, after they are detected.

1.4 Description of Dissertation by Chapter

This dissertation consists of seven chapters. In Chapter 1, advantages and history of multi-valued digital systems and the objectives of this research are given. Chapter 2 is to provide the necessary background for the reader to read this dissertation. Boolean algebra, switching Boolean algebra, binary vector switching algebra, Boolean derivative, series expansion, fault detection, and hazard detection in binary switching circuits are also given in Chapter 2.

In Chapter 3, multi-valued switching algebras and tabular minimization technique for minimizing multi-valued switching functions are introduced. Chapter 4 introduces the concepts of multi-valued partial derivative and multi-valued total differential. Computer method for computing multi-valued partial derivative and multi-valued total differential is given in Chapter 4. Chapter 5 of this dissertation describes a method of deriving a complete set of tests for detecting single faults on input lines (primary lines) and on internal lines of multi-valued combinational switching circuits using multi-valued partial derivative.

In Chapter 6, static hazard detection in asynchronous multi-level multi-valued combinational switching circuits using multi-valued total differential is considered, and Chapter 7 introduces a method for eliminating static hazards in multi-level multi-valued combinational switching circuits by using a new redundant circuitry concept.

CHAPTER II

BACKGROUND

2.1 Introduction

The purpose of this chapter is to provide the necessary background for the reader to read this dissertation. Materials presented in this chapter are extracted from S.C. Lee's two recently published books [18, Chapter 7, and 7, Chapters 1-3].

2.2 Boolean Algebra

Boolean algebra, first introduced by George Boole [19] constitutes an area of mathematics which is used in digital computers. Boolean algebra is widely used in the design and analysis of digital circuits and computers; it is the mathematical foundation of switching theory and logic design. Boolean algebra is introduced in this section, but first we introduce the AND and OR operations as follows:

<u>Definition 2.2.1</u>

Define the AND operation of Boolean algebra as:

$\mathrm{AND}(x,y) = \mathrm{Min}(x,y) = x \cdot y$

Definition 2.2.2

Define the OR operation of Boolean algebra as:

OR(x,y) = Max(x,y) = x+y

Definition 2.2.3

A Boolean algebra is an algebra $(B;\cdot,+,';0,1)$ consisting of Set B, AND (Boolean product) operation, OR (Boolean Sum) operation +, the NOT (complement) operation ' and the smallest and the largest elements 0 and 1. Let x,y, and z are elements in B, the following axioms hold for Boolean algebra.

Ia.	$x+y$ is in B	Ib.	$x\cdot y$ is in B
IIa.	x is in B	IIb.	x' is in B
IIIa.	$x+0 = x$	IIIb.	$x\cdot 1 = x$ (Null and universal elements)
IVa.	$x+x = x$	IVb.	$x\cdot x = x$ (Idempotent laws)
Va.	$x+y = y+x$	Vb.	$x\cdot y = y\cdot x$ (Commutative laws)
VIa.	$x+(y+z) = (x+y)+z$	VIb.	$x\cdot(y,z) = (x\cdot y)\cdot z$ (Associative laws)
VIIa.	$x+\bar{x} = 1$	VIIb.	$x\cdot\bar{x} = 0$ (Complement)
VIIIa.	$x+(x\cdot y) = x$	VIIIb.	$x\cdot(x+y) = x$ (Absorptive laws)
VIIIIa.	$x+(y\cdot z)=(x+y)\cdot(x+z)$	VIIIIb.	$x\cdot(y+z)=(x\cdot y)+x\cdot z$ (Distributive laws)

AND, OR and NOT operations are given in Table 2.2.1 for $B_2 = (\{0.1\};\ \cdot,\ +,\ ';\ 0,\ 1)$

TABLE 2.2.1

$\cdot$	0	1
0	0	0
1	0	1

(a) AND operation

+	0	1
0	0	1
1	1	1

(b) OR operation

'	
0	1
1	0

(c) NOT operation

Definition 2.2.4

If x is an element of B, then x is called a Boolean variable on B.

A Boolean function of Boolean algebra is defined as follows:

Definition 2.2.5

If $x_1, x_2, \ldots, x_n$ are Boolean variables in B, then $f(x_1, x_2, \ldots, x_n)$ is called a Boolean function and it can be constructed according to the following rules:

1. If $f(x_1, x_2, \ldots x_n)$ and $f_2(x_1, x_2, \ldots, x_n)$ are Boolean functions, then $f_1(x_1, x_2, \ldots, x_n))'$ is a Boolean function.
2. If $f_1(x_1, x_2, \ldots, x_n)$ and $f_2(x_1, x_2, \ldots, x_n)$ are Boolean functions, then $f_1(x_1, x_2, \ldots, x_n) + f_2(x_1, x_2, \ldots, x_n)$ and $f_1(x_1, x_2, \ldots, x_n) \cdot f_2(x_1, x_2, \ldots, x_n)$ are Boolean functions.
3. Any function that can be constructed by a finite numer of applications of the above rules is a Boolean function.

Every Boolean function can be written in either canonical sum-of-products form or canonical product-of-sums form. The canonical sum-of-products form is:

$$f(x_1,x_2,\ldots,x_n) = \sum_{i=1}^{\ell} f_i(x_1,x_2,\ldots,x_n)$$

and the canonical product-of-sums form is:

$$f(x_1,x_2,\ldots,x_n) = \prod_{i=1}^{\ell} f_i(x_1,x_2,\ldots,x_n)$$

where Σ is the Boolean summation, and Π is the Boolean product.

2.3 Switching Algebra

The two element Boolean algebra $B_2(\{0,1\};\cdot,+,';0,1)$ is known as switching algebra. Switching algebra is the mathematical foundation of the analysis and design of switching circuits that make up digital systems. Switching algebra contains the two extreme elements, the largest number represented by 1 and the smallest number represented by 0. Boolean functions defined on switching algebra is called switching functions. The properties of Boolean algebra of Section 2.2 are all applicable to switching algebra. Two very important theorems, which have many useful applications in regard to switching functions, are DeMorgan's theorem and Shannon's theorem.

Theorem 2.3.1 (DeMorgan's theorem)

(a) $(x_1+x_2+\ldots+x_n)' = x_1'\cdot x_2'\cdot\ldots\cdot x_n'$ (2.3.1)

(b) $(x_1\cdot x_2\cdot\ldots\cdot x_n)' = x_1'+x_2'+\ldots+x_n'$ (2.3.2)

DeMorgan's theorem does not give the relation between complementary functions. Shannon has suggested a generalization of the DeMorgan's theorem, which is described next.

<u>Theorem 2.3.2</u> (Shannon's theorem)

$$(f(x_1,x_2,\ldots,x_n;+,\cdot))' = f(x_1',x_2',\ldots,x_n',\cdot,+) \quad (2.3.3)$$

Shannon's theorem says that the complement of any function is obtained by replacing each variable by its complement and by interchanging the AND and OR operations.

The proofs of Theorems 2.3.1 and 2.3.2 can be found in most switching theory books, and thus are omitted.

2.4. Binary Vector Switching Algebra

The main objective of this section is to introduce binary-vector switching algebra, or simply vector switching algebra. Vector switching algebra was introduced for first time by Lee [6] in 1976. Vector switching algebra is a generalization of switching algebra which every element is represented by a binary vector. Moreover, the NOT or COMPLEMENTATION operation is extended to more general operation called the generalized complement, which includes the total complement (ordinary complement), the null complement (no complement), and newly introduced partial complement. For example, in ordinary Boolean algebra, the total complement of $X = (x_1,x_2,x_3)$ is as follows:

	$\bar{X}$(total complement)
000	111
001	110
010	101
011	100
100	011
101	010
110	001
111	000

In binary-vector switching algebra, the total complement can be generalized into more general operation. Let $X = (x_1, x_2, \ldots, x_n)$, the generalized complement of X is defined as follows:

Definition 2.4.1

The generalized complement X^P of a variable X is defined to be the element obtained from complementing the components of X according to the value of corresponding component of P; x_i is complemented or uncomplemented if p_i is 0 or 1, respectively, where x_i and p_i designate ith component of X and p.

For example the generalized complement of $X = (x_1, x_2, x_3)$ and P = 100 are as follows:

X	X^P
000	011
001	010
010	001
011	000
100	111
101	110
110	101
111	100

Now we shall introduce another operation, called the rotation operation, as follows:

Definition 2.4.2

The right $\overrightarrow{X}$ and left $\overleftarrow{X}$ rotation of X are defined as follows:

$$\overrightarrow{X} = (x_n, x_1, x_2, \ldots, x_{n-1})$$

and

$$\overleftarrow{X} = (x_2, x_3, \ldots, x_n, x_1)$$

when the arrow is omitted (i.e., $\tilde{X}$), it means that it can either be the right or the left rotation operation. $\overset{(m)}{\tilde{X}}$

$$\overset{(m)}{\tilde{X}} = \left.\tilde{\vdots}\right\} \text{ m times} \; X$$

Definition 2.4.3

The three operations +, ·, and ' plus the generalized complement and the rotation operation defined on it, is called a vector switching algebra.

The following are several basic properties of the generalized complement and the rotation operation.

PROPERTY 1

$X^0 = \bar{X}$ and $X^I = X$, where $0 = (0,\ldots,0)$ and $I = (1,\ldots,1)$

PROPERTY 2

(a) $P^P = I$

(b) $\overline{X^P} = X^{\bar{P}} = \bar{X}^P$

PROPERTY 3

$$X^{p^p} = (X^p)^p = X^{(p^p)} = X$$

PROPERTY 4

(a) $$\overline{(X_1^{p_1} + X_2^{p_2})} = \overline{X_1^{p_1}} \cdot \overline{X_2^{p_2}} = X_1^{\bar{p}_1} \cdot X_2^{\bar{p}_2} = \bar{X}_1^{p_1} \cdot \bar{X}_2^{p_2}$$

(b) $$\overline{(X_1^{p_1} \cdot X_2^{p_2})} = \overline{X_1^{p_1}} + \overline{X_2^{p_2}} = X_1^{\bar{p}_1} + X_2^{\bar{p}_2} = \bar{X}_1^{p_1} + \bar{X}_2^{p_2}$$

PROPERTY 5

(a) $$(X_1 + X_2)^p = (\bar{X}_1 \cdot \bar{X}_2)^{\bar{p}}$$

(b) $$(X_1 \cdot X_2)^p = (\bar{X}_1 + \bar{X}_2)^{\bar{p}}$$

PROPERTY 6

(a) $$(X_1^{p_1} + X_2^{p_2})^{p_3} = (X_1^{\bar{p}_1} \cdot X_2^{\bar{p}_2})^{\bar{p}_3}$$

(b) $$(X_1^{p_1} \cdot X_2^{p_2})^{p_3} = (X_1^{\bar{p}_1} + X_2^{\bar{p}_2})^{\bar{p}_3}$$

PROPERTY 7

(a) $$\widetilde{(X_1 \cdot X_2)} = \tilde{X}_1 \cdot \tilde{X}_2$$

(b) $$\widetilde{(X_1 + X_2)} = \tilde{X}_1 + \tilde{X}_2$$

PROPERTY 8

$$\widetilde{X^p} = \tilde{X}^{\tilde{p}}$$

PROPERTY 9

$$\text{(a)}\quad (X^{p})\,\underbrace{(X^{p})\,\ldots\,(X^{p})}_{n-1} = I \quad \text{if} \quad X = p$$
$$= 0 \quad \text{if} \quad X \neq p$$

$$\text{(b)}\quad X^{\bar{p}} + \underbrace{(X^{\bar{p}}) + \ldots + (X^{\bar{p}})}_{(n-1)} = 0 \quad \text{if} \quad X = p$$
$$= I \quad \text{if} \quad X \neq p$$

All the important theorems in ordinary switching algebra can now be generalized.

<u>Theorem 2.4.1</u> (Generalized DeMorgan's Theorem)

$$\text{(a)}\quad (X_1^{p_1} \cdot X_2^{p_2} \cdot \ldots \cdot X_m^{p_m})^{p} = (X_1^{\bar{p}_1} + X_2^{\bar{p}_2} + \ldots + X_m^{\bar{p}_m})^{\bar{p}} \tag{2.4.1}$$

$$\text{(b)}\quad (X_1^{p_1} + X_2^{p_2} + \ldots + X_m^{p_m})^{p} = (X_1^{\bar{p}_1} \cdot X_2^{\bar{p}_2} \cdot \ldots \cdot X_m^{\bar{p}_m})^{\bar{p}} \tag{2.4.2}$$

<u>Theorem 2.4.2</u> (Generalized Shannon's Theorem)

$$F^{p}(X_1^{p_1}, X_2^{p_2}, \ldots, X_m^{p_m}; +, \cdot) = F^{\bar{p}}(X_1^{\bar{p}_1}, X_2^{\bar{p}_2}, \ldots, X_m^{\bar{p}_m}, \cdot, +) \tag{2.4.3}$$

2.4.1 Canonical Form

Consider a general combinational circuit of Figure 2.4.1, where x_{ij}, $i = 1, \ldots, m$ and $j=1,\ldots,n$ are binary variables and f_k, $k=1,\ldots,n$ are binary functions. The n output functions of this circuit are:

$$f_1(x_{11},\ldots, x_{1n}, x_{21},\ldots, x_{2n},\ldots, x_{m1},\ldots,x_{mn})$$

$$\vdots$$

$$f_n(x_{11},\ldots, x_{1n}, x_{21},\ldots, x_{2n},\ldots, x_{m1},\ldots, x_{mn})$$

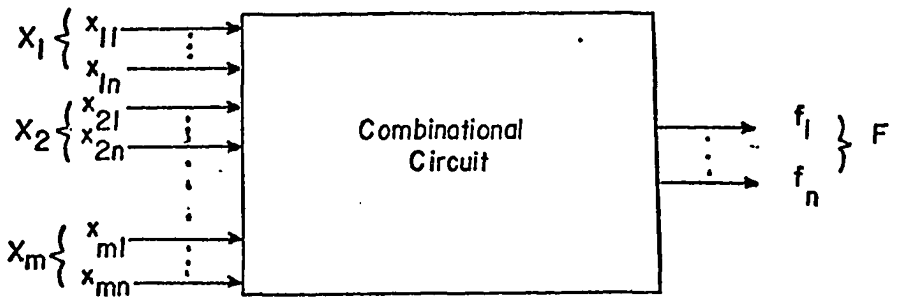


Figure 2.4.1: General Combinational Circuit

These n functions $f_1,\ldots, f_n$ can be described by a single vector output function of m n-tuple binary-vector variables. This single vector output function can be represented by the following canonical form.

<u>Theorem 2.4.3</u> (Canonical forms)

Let F be a vector switching function of m binary-vector variables $X_1, X_2, \ldots, X_m$. Then:

(a) The canonical sum-of-product form of F is

$$F(X_1, X_2, \ldots, X_m) = \Sigma\, F(P_1, \ldots, P_m)[X_1^{P_1}\ \overbrace{(X_1^{P_1})}^{} \ldots \overbrace{(X_1^{P_1})}^{(n-1)}] \ldots [X_m^{P_m}\overbrace{(X_m^{P_m})}^{} \ldots \overbrace{(X_m^{P_m})}^{(n-1)}] \qquad (2.4.4)$$

(b) The canonical product-of-sums form F is

$$F(X_1, X_2, \ldots, X_m) = \Pi\{F(P_1, \ldots, P_m) + [X_1^{P_1} + \overbrace{(X_1^{P_1})}^{} + \ldots + \overbrace{(X_1^{P_1})}^{(n-1)}] + \ldots + [X_m^{P_m} + \overbrace{(X_m^{P_m})}^{} + \ldots + \overbrace{(X_m^{P_m})}^{(n-1)}]$$

$$(2.4.5)$$

where $P_1, \ldots, P_m$ take on values $0, 1, \ldots, 2^n-1$ in binary form. The proof of theorem 2.4.3 is given by Lee [7].

<u>Example 2.3.1</u>

Consider the truth tuble Table 2.4.1 of the vector switching $F(X_1, X_2)$. Find the canonical sum-of-products of $F(X_1, X_2)$.

TABLE 2.4.1

X_1	X_2	F
0	0	1
0	1	3
1	2	2
2	3	1
Otherwise		0

(a) Truth Table

X_1	X_2	F
00	00	01
00	01	11
01	10	10
10	11	01
Otherwise		00

(b) Binary-coded truth table of (a)

$$F(X_1,X_2) = (01)\ X_1^{00}\overbrace{(X_1^{00})}\ X_2^{00}\overbrace{(X_2^{00})} + (11)\ X_1^{00}\overbrace{(X_1^{00})}\ X_2^{01}\overbrace{(X_2^{01})}$$

$$+ (10)\ X_1^{01}\overbrace{(X_1^{01})}\ X_2^{10}\overbrace{(X_2^{10})} + (01)X_1^{10}\overbrace{(X_1^{10})}X_2^{11}\overbrace{(X_2^{11})}$$

Many other properties and theorems of vector switching algebra and its applications to logic desing are given by Lee [7].

2.5 Boolean Derivative

Boolean derivative (partial derivative of two-valued logic) has been studied in two papers due to Akers [2] and to Sellers, Hsiao and Bearson [3]. In these two papers, various properties of Boolean derivative are introduced. The Boolean derivative and its properties are introduced in this section. But first let us introduce the EXCLUSIVE-OR operation of switching functions as follows:

<u>Definition 2.5.1</u>

Define the EXCLUSIVE-OR operation of two-valued switching algebra as:

$$\text{EX-OR}(x,y) = x \oplus y = xy' + x'y$$

This EX-OR operation satisfies the commutative laws, the associative laws and the distributive laws that are described in Section 2.2 which they make the EX-OR operation is a functionately complete operation. The Boolean derivative is defined as follows:

<u>Definition 2.5.2</u>

If f is a switching function that has one output and n input variables $x_1,\ldots,x_n$, then the Boolean derivative of f with respect to the variable x_i is:

$$\frac{\partial f(x_1,\ldots,x_i,\ldots,x_n)}{\partial x_i} = f(x_1,\ldots,x_i,\ldots,x_n) \oplus f(x_1,\ldots,x_i',\ldots,x_n) \tag{2.5.1}$$

Definition 2.5.2 is equivalent to the following definition.

<u>Definition 2.5.2'</u>

$$\frac{\partial f(x_1,\ldots,x_i,\ldots,x_n)}{\partial x_i} = f(x_1,\ldots,1,\ldots,x_n) \oplus f(x_1,\ldots,0,\ldots,x_n) \tag{2.5.2}$$

Example 2.5.1

Find the Boolean derivative of the following switching function with respect to the variable x_1.

$$f(x_1,x_2,x_3,x_4) = (x_1 + x_2)(x_3' + x_2')(x_1' + x_4')$$

$$\begin{aligned}\frac{\partial f(x_1,x_2,x_3,x_4)}{\partial x_1} &= (1 + x_2)(x_3' + x_2')(0 + x_4') \oplus (0 + x_2) \\ &\cdot (x_3' + x_2')\ (1 + x_4') \\ &= (x_3' + x_2')\ x_4' \oplus x_2\ (x_3' + x_2') \\ &= x_2'\ x_4' + x_2\ x_3'\ x_4\end{aligned}$$

The properties of the Boolean derivative are given below.

PROPERTY 1

$$\frac{\partial a}{\partial x_i} = 0 \qquad \text{where a is a Boolean constant}$$

PROPERTY 2

$$\frac{\partial (af)}{\partial x_i} = a\ \frac{\partial f}{\partial x_i}$$

PROPERTY 3

$$\frac{\partial \bar{f}}{\partial x_i} = \frac{\partial f}{\partial x_i}$$

PROPERTY 4

$$\frac{\partial}{\partial x_i}\ (\frac{\partial f}{\partial x_i}) = 0$$

PROPERTY 5

$$\frac{\partial f}{\partial x_i'} = \frac{\partial f}{\partial x_i}$$

PROPERTY 6

$$\frac{\partial}{\partial x_i}\left(\frac{\partial f}{\partial x_j}\right) = \frac{\partial}{\partial x_j}\left(\frac{\partial f}{\partial x_i}\right)$$

PROPERTY 7

$$\frac{\partial (f\cdot g)}{\partial x_i} = f\,\frac{\partial g}{\partial x_i} \oplus g\,\frac{\partial f}{\partial x_i} \oplus \frac{\partial f}{\partial x_i}\cdot\frac{\partial g}{\partial x_i}$$

PROPERTY 8

$$\frac{\partial (f+g)}{\partial x_i} = \bar{f}\,\frac{\partial g}{\partial x_i} \oplus \bar{g}\,\frac{\partial f}{\partial x_i} \oplus \frac{\partial f}{\partial x_i}\cdot\frac{\partial g}{\partial x_i}$$

PROPERTY 9

$$\frac{\partial (f\oplus g)}{\partial x_i} = \frac{\partial f}{\partial x_i} \oplus \frac{\partial g}{\partial x_i}$$

<u>Example 2.5.2</u>

Repeat Example 2.5.1 using Property 7.

$$f\cdot g = (x_1+x_2)(x_3'+x_2')(x_1'+x_4')$$

Let $f = x_1+x_2$ and $g = (x_3'+x_2')(x_1'+x_4')$

Also let $g = g_1\cdot g_2 = (x_3'+x_2')(x_1'+x_4')$

i.e., $g_1 = x_3'+x_2'$ and $g_2 = x_1'+x_4'$

We need to compute $\frac{\partial f}{\partial x_1}$ and $\frac{\partial g}{\partial x_1}$ in order to compute $\frac{\partial (f\cdot g)}{\partial x_1}$ using Property 7.

$$\frac{\partial f}{\partial x_1} = (1+x_2) \oplus (0+x_2) = x_2'$$

In order to compute $\frac{\partial g}{\partial x_1}$, we need first to compute $\frac{\partial g_1}{\partial x_1}$ and and $\frac{\partial g_2}{\partial x_2}$:

$$\frac{\partial g_1}{\partial x_1} = 0, \ \frac{\partial g_2}{\partial x_1} = (0+x_4') \oplus (1+x_4') = x_4$$

then $\frac{\partial g}{\partial x_1} = \frac{\partial (g_1 \cdot g_2)}{\partial x_1} = g_1 \frac{\partial g_2}{\partial x_1} \oplus g_2 \frac{\partial g_1}{\partial x_1} \oplus \frac{\partial g_1}{\partial x_1} \cdot \frac{\partial g_2}{\partial x_1}$

$$= x_4(x_3' + x_2')$$

Finally,

$$\frac{\partial (f \cdot g)}{\partial x_1} = f \frac{\partial g}{\partial x_1} \oplus g \frac{\partial f}{\partial x_1} \oplus \frac{\partial f}{\partial x_1} \cdot \frac{\partial g}{\partial x_1}$$

$$= x_4(x_1+x_2)(x_3'+x_2') \oplus x_2'(x_3'+x_2')(x_1'+x_4') \oplus x_2'x_4(x_3'+x_2')$$

$$= x_2' x_4' + x_2 x_3' x_4$$

The same result is obtained as in Example 2.5.1.

2.6 Series Expansion

Since we have defined a process similar to ordinary differentiation, i.e., the invention of Boolean partial derivative it becomes very natural to ask if it is possible to expand Boolean functions. It is found by Akers [2] that is is possible that the Boolean functions can be expanded in MacLaurin series and Taylor series forms.

Theorem 2.6.1

The MacLaurin series expansion form of a Boolean switching function is:

$$f(\mathbf{x},\mathbf{y}) = \sum_{\mathbf{e}} \left(\frac{\partial f}{\partial \mathbf{y}^{\mathbf{e}}}\right)_{\mathbf{y}=\mathbf{0}} \mathbf{y}^{\mathbf{e}}, \quad 0 \leq \mathbf{e} \leq 2^{p}-1 \qquad (2.6.1)$$

where, Σ is the Boolean EX-OR summation

$y^{e} = 1$ if $e=0$

$y^{e} = y$ if $e=1$

$\mathbf{x} = (x_1, x_2, \ldots, x_n)$

$\mathbf{y} = (y_1, y_2, \ldots, y_p)$

$\mathbf{e} = (e_1, e_2, \ldots, e_p)$

Theorem 2.6.2

The Taylor series expansion form of a Boolean switching function is

$$f(\mathbf{x},\mathbf{y}) = \sum_{\mathbf{e}} \left(\frac{\partial f}{\partial \mathbf{y}^{\mathbf{e}}}\right)_{\mathbf{y}=\mathbf{h}} (\mathbf{y} \oplus \mathbf{h})^{\mathbf{e}}, \quad 0 \leq \mathbf{e} \leq 2^{p}-1 \qquad (2.6.2)$$

where, $\mathbf{h} = (h_1, h_2, \ldots, h_p)$

$(y \oplus h) = y'$ if $h = 1$

$(y \oplus h) = y$ if $h = 0$

$(y \oplus h)^{e} = 1$ if $e = 0$

$(y \oplus h)^{e} = (y \oplus h)$ if $e = 1$

h is the assignment to the y's, i.e., one of the 2^p vertices of the y cube. Notice, if h=0, then the Taylor series expansion becomes identical to MacLaurin series expansion. The proofs of theorems 2.6.1 and 2.6.2 are given by Akers [2].

Example 2.6.1

Find the MacLaurin series expansion of the following Boolean switching function.

$$f(x_1,x_2,y_1,y_2) = x_1y_1 + y_1'y_2 + x_2'y_2'$$

The MacLaurin series expansion for p = 2 is:

$$f(x_1,x_2,y_1,y_2) = f(x_1,x_2,0,0) \oplus \frac{\partial f}{\partial y_1}\bigg|_{\substack{y_1=0\\ y_2=0}} y_1 \oplus \frac{\partial f}{\partial y_2}\bigg|_{\substack{y_1=0\\ y_2=0}} y_2$$

$$\oplus \left(\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_2}\right)\right)\bigg|_{\substack{y_1=0\\ y_2=0}} y_1y_2$$

To obtain the series expansion, we need to compute the following three partial derivatives.

(1) $\frac{\partial f}{\partial y_1}$ (2) $\frac{\partial f}{\partial y_2}$ (3) $\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right)$

(1) $\frac{\partial f}{\partial y_1} = (x_1+x_2'y_2') \oplus (y_2 + x_2'y_2') = x_1'y_2 + x_1x_2y_2'$

$$\frac{\partial f}{\partial y_1}\bigg|_{\substack{y_1=0\\ y_2=0}} = x_1x_2$$

(2) $\frac{\partial f}{\partial y_2} = (x_1y_1+y_1') \oplus (x_1y_1+x_2') = x_2y_1'+x_1'x_2'y_1$

$$\left.\frac{\partial f}{\partial y_2}\right|_{\substack{y_1=0\\y_2=0}} = x_2$$

(3) $\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right) = \cdot\ x_1' \oplus x_1x_2 = x_1'+x_2$

$$\left.\left(\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right)\right)\right|_{\substack{y_1=0\\y_2=0}} = x_1'+x_2$$

Then, the MacLaurin series expansion is:

$$f(x_1,x_2,y_1,y_2) = x_2' \oplus x_1x_2y_1 \oplus x_2y_2 \oplus (x_1'+x_2)\ y_1y_2$$

Example 2.6.2

Find the Taylor series expansion for h=10 of the function of Example 2.6.1.

The Taylor series expansion for p=2 and h=10 is:

$$f(x_1,x_2,y_1,y_2) = f(x_1,x_2,1,0) \oplus \left.\frac{\partial f}{\partial y_1}\right|_{\substack{y_1=1\\y_2=0}} (y_1 \oplus 1)$$

$$\oplus \left.\frac{\partial f}{\partial y_2}\right|_{\substack{y_1=0\\y_2=0}} (y_2 \oplus 0) \oplus \left.\left(\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right)\right)\right|_{\substack{y_1=1\\y_2=0}}$$

$$(y_1 \oplus 1)(y_2 \oplus 0) = f(x_1,x_2,1,0) \oplus \left.\frac{\partial f}{\partial y_1}\right|_{\substack{y_1=1\\y_2=0}}$$

$$y_1' \oplus \left.\frac{\partial f}{\partial y_2}\right|_{\substack{y_1=1\\y_2=0}} y_2 \oplus \left.\left(\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right)\right)\right|_{\substack{y_1=1\\y_2=0}} y_1'\ y_2$$

But,

(1) $$\left.\frac{\partial f}{\partial y_1}\right|_{\substack{y_1=1\\ y_2=0}} = x_1x_2$$

(2) $$\left.\frac{\partial f}{\partial y_2}\right|_{\substack{y_1=1\\ y_2=0}} = x_1' x_2'$$

(3) $$\left.\left(\frac{\partial}{\partial y_2}\left(\frac{\partial f}{\partial y_1}\right)\right)\right|_{\substack{y_1=1\\ y_2=0}} = x_1' + x_2$$

Then, the Taylor series expansion for h=10 is:

$$f(x_1,x_2,y_1,y_2) = (x_1+x_2') \oplus x_1x_2y_1' \oplus x_1'x_2'y_2 \oplus (x_1'+x_2)y_1'y_2$$

It can be noticed that there are two more expansions for the function $f(x_1,x_2,y_1,y_2)$ when h=01 and h=11 which can be computed in similar way.

2.7 Fault Detection

The fault testing and diagnosis of binary combinational circuits is an important part of the system design, because any binary combinational circuits may be disabled by an error. In this section, we will use the concept of Boolean partial derivative for deriving the fault detection tests. The application of these tests to a circuit, it can determine whether the circuit is faulty or it is fault-free. From the definition of Boolean partial derivative Definition 2.5.2, it is easy to see the following:

$$\frac{\partial f}{\partial x_i} = 1, \text{ when } f(x_1,\ldots,x_{i-1},1,x_{i+1},\ldots,x_n) \neq f(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_n)$$

$$\frac{\partial f}{\partial x_i} = 0, \text{ when } f(x_1,\ldots,x_{i-1},1,x_{i+1},\ldots,x_n) = f(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_n)$$

Thus, we can state the following two conditions:

(1) if $\frac{\partial f}{\partial x_i} = 0$, then an error in x_i does not cause an error in $f(x)$, thus this error is undetectable.

(2) if $\frac{\partial f}{\partial x_i} = 1$, then an error in x_i will always cause an error in $f(x)$.

Condition (2) above is used to find all s-a-0 and s-a-1 tests in which these tests can be applied to the circuit to detect s-a-0 and s-a-1 faults on line i. But, sometimes we like to find the s-a-0 tests and s-a-1 tests separately, this can be done by changing condition (2) into the following two conditions:

(3) if $x_i \frac{\partial f}{\partial x_i} = 1$, then s-a-0 error in x_i is exist

(4) if $\bar{x}_i \frac{\partial f}{\partial x_i} = 1$, then s-a-1 error in x_i is exist

Let T be the set of s-a-0 and s-a-1 tests obtained from condition (2), to be the set of s-a-0 tests obtained from condition (3) and T_1 be the set of s-a-1 tests obtained from condition (4), then we have

$$T = T_0 + T_1$$

Two or more errors at the input of a binary circuit whose output function $f(x_1,\ldots,x_i,x_{i+1},\ldots,x_{i+\ell},\ldots,x_n)$ can be detected by using the following multiple Boolean derivative.

<u>Definition 2.7.1</u>

The multiple Boolean derivative of the function $f(x_1,\ldots,x_i,x_{i+1},\ldots,x_{i+\ell},\ldots,x_n)$ is defined as follows:

$$\frac{\partial f(x_1,\ldots,x_i,x_{i+1},\ldots,x_{i+\ell},\ldots,x_n)}{\partial(x_i x_{i+1}\cdots x_{i+\ell})} = f(x_1,\ldots,x_i,x_{i+1},\ldots, x_{i+\ell},\ldots,x_n) \oplus f(x_1,\ldots,\bar{x}_i, \bar{x}_{i+1},\ldots,\bar{x}_{i+\ell},\ldots x_n) \tag{2.7.1}$$

If the given circuit is to be tested against an error on one of its internal lines, then the output function f must be written as the function of the input variables $x_1,\ldots,x_n$ and the function g_i, i.e., $f(x_1,\ldots,x_n,g_i)$, where g_i is the output function of that part of the circuit where the error exists and it is a function of the input variables $x_1,\ldots,x_n$. Therefore, conditions (2), (3) and (4) can be rewritten as follows:

(5) if $\dfrac{\partial f(x_1,\ldots,x_n,g_i)}{\partial g_i} = 1$, then an error in g_i exists

(6) if $g_i(x_1,\ldots,x_n)$ $\dfrac{\partial f(x_1,\ldots,x_n,g_i)}{\partial g_i} = 1$, then s-a-0 error on line i exists

(7) if $\bar{g}_i(x_1,\ldots,x_n)$ $\dfrac{\partial f(x_1,\ldots,x_n,g_i)}{\partial g_i} = 1$, then s-a-1 error on line i exists

<u>Example 2.7.1</u>

Let us consider a fault on line i of the circuit shown in Figure 2.7.1.

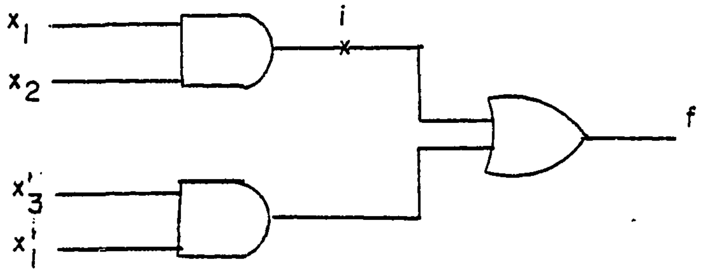


Figure 2.7.1: The Logic Circuit of Example 2.7.1

Since,

$$f(x_1,x_2,x_1) = x_1x_2+x_1'x_3'$$

We have

$$f(x_1,x_3,g_i) = g_i + x_1'x_3'$$

$$\begin{aligned}\frac{\partial f(x_1,x_3,g_i)}{\partial g_i} &= (g_i+x_1'x_3') \oplus (\bar{g}_i+x_1'x_3') \\ &= (x_1x_2 + x_1'x_3') \oplus (\overline{x_1x_2} + x_1'x_3') \\ &= x_1 + x_3\end{aligned}$$

$$g_i\ (x_1,x_2) = x_1x_2$$

and

$$\bar{g}_i(x_1,x_2) = x_1' + x_2'$$

By using conditions (6) and (7), we obtain:

$$g_i(x_1,x_2)\ \frac{\partial f(x_1,x_3,g_i)}{\partial g_i} = x_1x_2$$

and

$$\bar{g}_i(x_1,x_2)\ \frac{\partial f(x_1,x_2,g_i)}{\partial g_i} = x_1x_2'+x_1'x_3+x_2'x_3 = 1$$

Thus, s-a-0 tests $T_0 = \{110, 111\}$ and s-a-1 tests $T_1 = \{001, 011, 100, 101\}$.

2.8 Hazard Detection

Whenever an input variable of a binary combinational circuit is allowed to change, it is possible to have some undesirable operating situations if this variable and its complement are both present in the realization. Since this variable x_i and its complement x_i' may not change exactly at the same time and thus the circuit may momentarily give some incorrect outputs to occur. This incorrect undesirable behavior of the output of the circuit is called hazard. There are two types of hazards that cause incorrect output behavior, they are called the static hazard and the dynamic hazard in which they are defined in the following two definitions respectively.

Definition 2.8.1

A combinational asychronous logic circuit contains static hazard for a single input change if and only if (1) the outputs before and after the change are equal (2) during the change a spurious pulse may appear on the output.

If the outputs before and after changing the input are both 1, with an incorrect output 0 in between, i.e., the output sequence is 1-0-1, then the hazard is a static 1 hazard. If the outputs before and after changing the input are both 0, with an incorrect output 1 in between, i.e., the output sequence is 0-1-0, then the hazard is a static 0 hazard.

Definition 2.8.2

A combinational asynchronous logic circuit contains dynamic hazard for a single input change if and only if:
(1) the outputs before and after the change are different.
(2) During the change, the output changes three or more times.

If the outputs before and after changing the input are different, and the output changes three times instead of once and passes through an additional temporary sequence of 01 or 10 in going to the final output, i.e., the output sequence is either 1-0-1-0 or 0-1-0-1, then the hazard is a dynamic hazard.

Sheng [20] proved that the static 0 hazard does not exist in a two layer sum-of-product form realization. Thus, if such a circuit is free of static 1 hazard, it is free of all static hazards. Sheng [20] also proved that static 1 hazard does not exist in a two layer product-of-sums form realization. Dynamic hazard is not present in sum-of-product and product-of-sum form gate realizations when only single input change is considered, this case is also proved by Sheng [20].

Example 2.8.1

Study Static 1 hazard of the circuit of Figure 2.7.1. The function f of the circuit of Figure 2.7.1 is:

$$f(x_1,x_2,x_1) = x_1x_2 + x_1' x_3'$$

The Karnaugh map of f is shown in Figure 2.8.1.

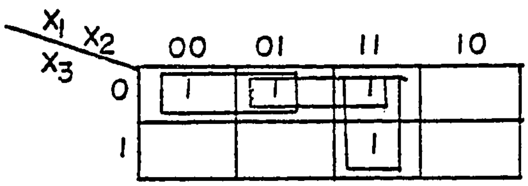


Figure 2.8.1: Karnaugh Map of $f = x_1x_2 + x_1' x_3'$

Since only the variable x_1 and its complement x_1' are present in the realization thus Static 1 hazard is only due to the presence of x_1 and x_1'. The transition from the input state (010) to the input state (110) causes the Static 1 hazard to be present which makes the output sequence is 1-0-1. This hazard can be eliminated by adding the prime implicant x_2x_3' to the circuit and this is indicated by the dashed square on the Karnaugh map of Figure 2.8.1.

CHAPTER III

MULTI-VALUED SWITCHING SYSTEMS

3.1 Introduction

An increasing interest in research has been shown in the subject of multi-valued switching systems. The development in this area has been active for quite some time [8-17]. Most definitions and theorems of two-valued switching system (binary switching system) can be extended to the m-valued switching systems. However, the multi-valued switching systems in general are far more difficult to analyze than the two-valued switching system because the multi-valued switching systems in general are not symmetric. For example, 4-valued switching function is symmetric while 6-valued switching function is not symmetric.

Just as two-valued switching system is described by two-valued switching algebra (Boolean switching algebra), multi-valued switching systems can be described by multi-valued switching algebra. Multi-valued switching algebra is the mathematical foundation of the analysis and design of multi-valued switching circuits that make up multi-valued digital circuits. Multi-valued switching algebra is an algebra such that functions of arbitrary complexity may be represented in terms of simple algebraic combinations of the basic functions. In addition, the choice of basic functions and the algebra permits the development of a technique to simplify in some useful sense the complexity of the functional representations. The multi-valued switching algebra and its basic functions (basic operations) are discussed in next section.

3.2 Multi-Valued Switching Algebra

It is potentially advantageous to have multi-valued switching algebra that can be used to any switching function regardless of the choice of the value of the switching function. Multi-valued switching algebra is said to be potentially implementable for practical applications, or simply, potentially implementable if it possesses the following three properties:

(a) A set of practically implementable basic functions which constitute a functionally complete set for realizing any switching function defined on the algebra.

(b) Canonical forms.

(c) Well-defined function minimization techniques.

Among the multi-valued algebras published in the literature, it is found that two algebras, one due to Su and Sarris [12] and the other due to Vranesic, Lee, and Smith [11] are potentially implementable. These two algebras will be considered in the next two subsections. For convenience, these two multi-valued switching algebras [12] and [11] will be referred to as Algebra A and Algebra B, respectively.

3.2.1 Algebra A

The multi-valued switching algebra introduced by Su and Sarris [12] (Algebra A) is potentially implementable and it is defined as follows:

(1) It contains a set of variables $(x,y,z,\ldots)$ which can assume m logic values from the set $Q=\{0,1,\ldots,m-1\}$, where $0<1<\ldots<m-1$.

(2) There exists an equivalence (=) operation, that is

$x=x$

if $x=y$, then $y=x$,

if $x=y$ and $y=z$, then $x=z$.

(3) It has the following basic operations:

(a) Two-element operations

$$x+y = \max(x,y)$$

$$x\cdot y = \min(x,y)$$

where $\max(x,y)$ and $\min(x,y)$ indicate the highest and the lowest values of (x,y), respectively.

(b) Let $c \in \{0,1,\ldots,m-1\}$, the complement of c, denoted by $\bar{c}$, is defined as

$$\bar{c} = (m-1) - c$$

(c) Define a variable as

$$\begin{aligned} {}^{a,b}x &= m-1, \text{ if } a \le x \le b \\ &= 0 \quad \text{Otherwise} \end{aligned}$$

where $x^{a,b} = x^{a,a} + x^{a+1,a+1} + \ldots + x^{b,b}$ (3.2.1)

where $a,b \in \{0,1,\ldots,m-1\}$ and $a \leq b$. Note that $x^{a,b}$ maps a m-valued variable x into binary space $\{0,m-1\}$. $x^{a,b}$ maps the values of $x=a, a+1,\ldots, b-1,b$ into m-1 and the values $x=0,1,\ldots,a-1$ and $x=b+1, b+2,\ldots,m-1$ to 0. After the mapping the variable $x^{a,b}$ can be treated just like a Boolean variable. The complement of $x^{a,b}$, denoted by $\overline{x^{a,b}}$, is defined as:

$$\overline{x^{a,b}} = 0 \quad \text{if } a \leq x \leq b$$

$$= m-1 \quad \text{Otherwise}$$

which is also two-valued. It should be noted that

$$\overline{x^{a,b}} = x^{0,a-1} + x^{b+1,\,m-1} \qquad (3.2.2)$$

where $x^{0,a-1} = 0$ for $a=0$ $x^{b+1,m-1} = 0$ for $b=m-1$.

AND, OR and NOT operations for 3-valued functions are given in Table 3.2.1

TABLE 3.2.1

·	0	1	2
0	0	0	0
1	0	1	1
2	0	1	2

(a) 3-Valued AND operation

+	0	1	2
0	0	1	2
1	1	1	2
2	2	2	2

(b) 3-Valued OR operation

'	
0	2
1	1
2	0

(c) 3-Valued NOT operation

(4) The two element operations defined above obey the following properties.

Ia.	$x+0 = x$	Ib.	$x \cdot m-1 = x$ (Null and universal elements)
IIa.	$x+x = x$	IIb.	$x \cdot x = x$ (Idempotent laws)
IIIa.	$x+y = y+x$	IIIb.	$x \cdot y = y \cdot x$ (Commutative laws)
IVa.	$x+(y+z) = (x+y)+z$	IVb.	$x \cdot (y \cdot z) = (x \cdot y) \cdot z$ (Associative laws)
Va.	$x+\bar{x} = m-1$	Vb.	${}^{a,b}x \; {}^{\overline{a,b}}x \doteq 0$ (Complement)
VIa.	$\bar{\bar{x}} = x$		(Involution)

It has been shown [12] that DeMorgan's theorem holds for multi-valued logic using Algebra A.

<u>Theorem 3.2.1</u> (DeMorgan's theorem)

$$\text{(a)} \quad ({}^{a_1,b_1}x_1 \cdot {}^{a_2,b_2}x_2 \cdots {}^{a_n,b_n}x_n)' = {}^{\overline{a_1,b_1}}x_1 + {}^{\overline{a_2,b_2}}x_2 + \dots + {}^{\overline{a_n,b_n}}x_n \tag{3.2.3}$$

$$\text{(b)} \quad ({}^{a_1,b_1}x_1 + {}^{a_2,b_2}x_2 + \dots + {}^{a_n,b_n}x_n)' = {}^{\overline{a_1,b_1}}x_1 \cdot {}^{\overline{a_2,b_2}}x_2 \cdots {}^{\overline{a_n,b_n}}x_n \tag{3.2.4}$$

where $a_1,\dots,a_n$ and $b_1,\dots,b_n \; \varepsilon \{0,1,\dots,m-1\}$

<u>Theorem 3.2.2</u> (Shannon's theorem)

$$(f(k_1 \cdot {}^{a_1,b_1}x_1, \dots, k_n \; {}^{a_n,b_n}x_n \,;\, +, \cdot))' = f(k_1 \cdot \overline{{}^{a_1,b_1}x_1}, \dots, k_n \cdot \overline{{}^{a_n,b_n}x_n} \,;\, \cdot, +) \tag{3.2.5}$$

The proofs of theorems 3.2.1 and 3.2.2 are given by [12].

It has been shown [12] that the sum, product, and complement operations of this algebra are a set of functionally complete operations, and any multi-valued switching functions can be expressed by the sum-of-product canonical form:

$$f(x_1,\ldots,x_n) = \sum_{\ell} c_\ell \cdot x_1^{a_{\ell 1},b_{\ell 1}} \cdot x_2^{a_{\ell 2},b_{\ell 2}} \cdots x_n^{a_{\ell n},b_{\ell n}} \qquad (3.2.6)$$

The complement of f is

$$\bar{f}(x_1,\ldots,x_n) = \sum_{\ell} (\bar{c}_\ell + \overline{x_1^{a_{\ell 1},b_{\ell 2}}} + \overline{x_2^{a_{\ell 2},b_{\ell 2}}} + \ldots + \overline{x_n^{a_{\ell n},b_{\ell n}}}) \qquad (3.2.7)$$

where $\bar{c}_\ell = (m-1) - c_\ell$ and $c_\ell \in \{0,1,\ldots,m-1\}$.

The set of basic operations can be realized by the seven basic components shown in Figure 3.2.1 where $K \in \{1,2,\ldots,m-1\}$. The electronic implementations of the K.-gate, K.-AND-gate, K-OR-gate and the digitizer remain to be investigated, but AND-gate, OR gate, and the inverter can be implemented using conventional, AND, OR, and NOT gates, respectively.

Example 3.2.1: The sum-of-products canonical form of the multi-valued switching function described by the truth table of Table 3.2.2 is:

$$\begin{aligned} f(x,y) = {} & 1\cdot[x^{4,4}y^{0,0} + x^{4,4}y^{1,1} + x^{4,4}y^{2,2} + x^{4,4}y^{3,3} + x^{4,4}y^{4,4}] \\ & + 2\cdot[x^{0,0}y^{2,2} + x^{0,0}y^{3,3} + x^{1,1}y^{3,3} + x^{1,1}y^{2,2}] \\ & + 3\cdot[x^{0,0}y^{0,0} + x^{0,0}y^{4,4}] \end{aligned} \qquad (3.2.8)$$

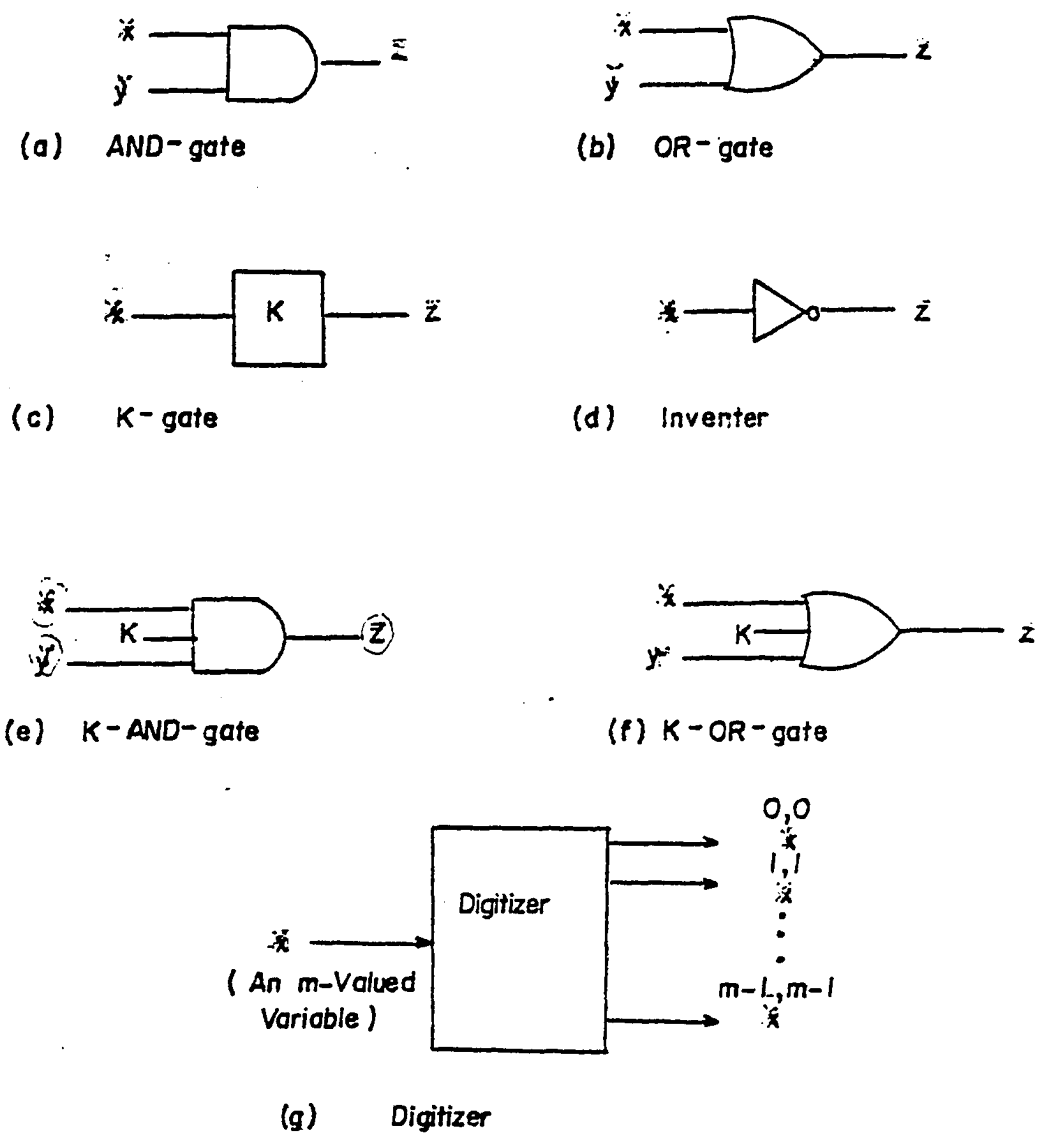


Figure 3.2.1: Seven basic circuit components of the multi-valued switching system derived from algebra [5].

The multi-valued switching function can be minimized by either using m-valued map which is similar to the Karnaugh map or by using the tabular method which is known as Quine-McCluskey method. For example, the function of Equation (3.2.8) can be minimized by using a 5-valued map which is shown in Figure 3.2.2. From this map it is found that the minimized function is:

$$f(x,y) = 1 \cdot x^{4,4} + 2 \cdot x^{0,1} y^{2,3} + 3 \cdot x^{0,0} y^{\overline{1,3}} \qquad (3.2.9)$$

TABLE 3.2.2

TRUTH TABLE OF THE FUNCTION OF EXAMPLE 3.2.1

x	0	0	0	1	1	0	4	4	4	4	4	Otherwise
y	0	2	3	3	2	4	0	1	2	3	4	
	3	2	2	2	2	3	1	1	1	1	1	0

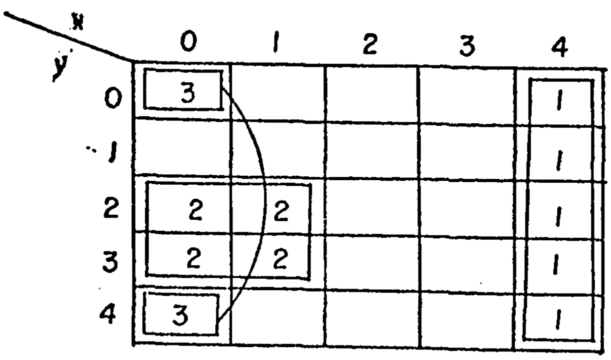


Figure 3.2.2: 5-valued map of the truth table of Table 3.2.2.

Example 3.2.2 Find K-AND-OR realization of Equation 3.2.9)

In order to get K-AND-OR realization of Equation (3.2.4), the function f(x,y) of Equation (3.2.9) must be rewritten such that all variable in its terms must be in the form of $x^{a,a}$. By applying Equation (3.2.1) to f(x,y), we obtain

$$f(x,y) = 1\cdot x^{4,4} + 2\cdot(x^{0,0} + x^{1,1})(y^{2,2} + y^{3,3}) + 3\cdot x^{0,0}(y^{0,0} + y^{4,4})$$

$$= 1\cdot x^{4,4} + 2\cdot x^{0,0}y^{2,2} + 2\cdot x^{0,0}y^{3,3} + 2\cdot x^{1,1}y^{2,2} + 2\cdot x^{1,1}y^{3,3}$$

$$+ 3\cdot x^{0,0}y^{0,0} + 3\cdot x^{0,0}y^{4,4} \tag{3.2.10}$$

K-AND-OR realization of Equation (3.2.10) is shown in Figure (3.2.3).

3.2.2 Algebra B

This algebra [11] is defined the same as algebra A except (3)b and (3)c are replaced by the following operations:

(1) m unary "Inverter" Operations:

$$\dot{x}^K = K \quad \text{if } f\ x = 0$$
$$= 0 \quad \text{otherwise}$$

(2) m unary "Clockwise Cycling" Operations:

$$\overset{M}{\underset{x}{\rightarrow}} = (x + M) \bmod m$$

where $M \in Q$.

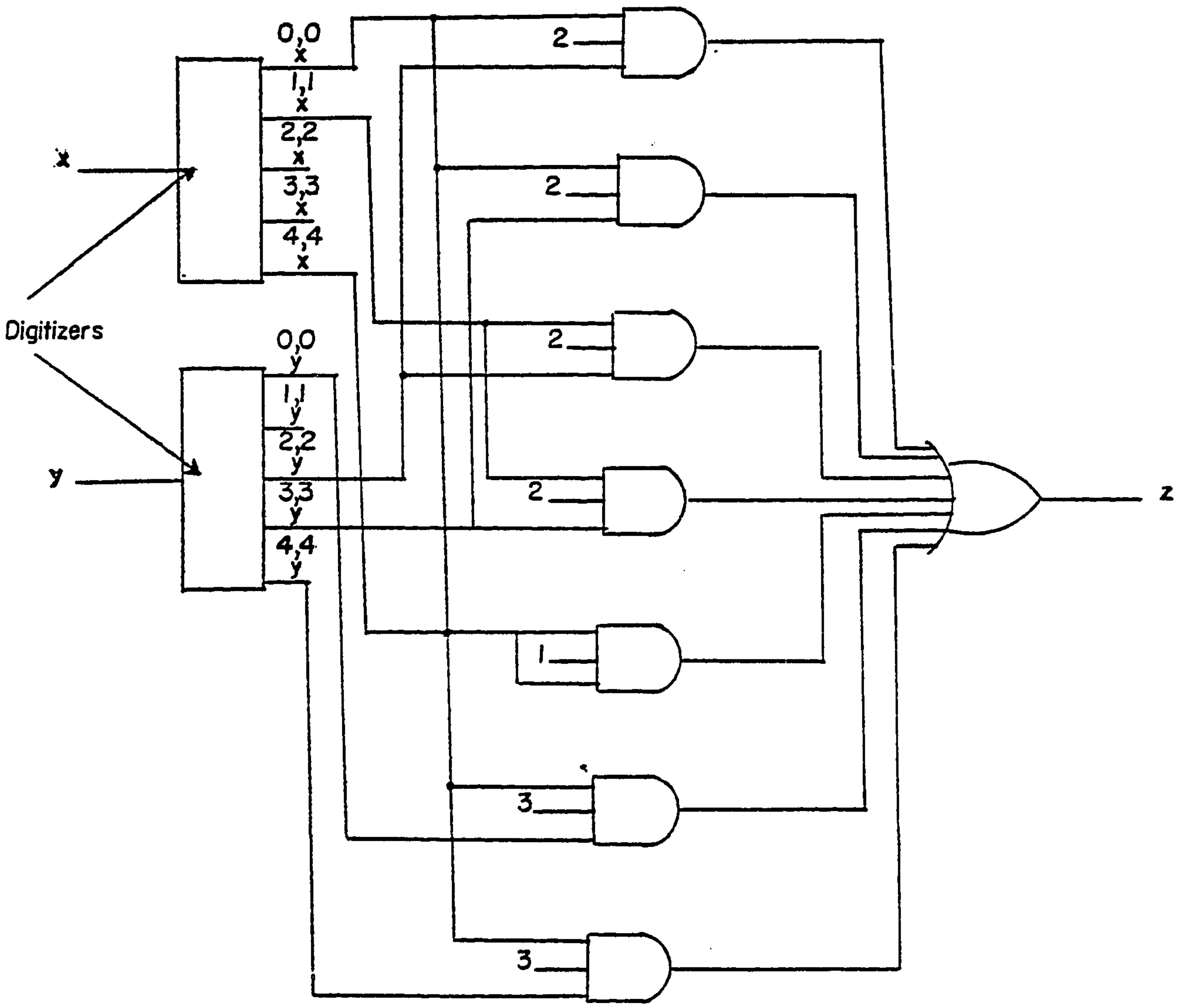


Figure 3.2.3: K-AND-OR Realization of Equation (3.2.10).

m unary inverter and m Unary clockwise cycling operations for m=3 is given in the following table.

TABLE 3.2.3

x^k	0	1	2
0	0	1	2
1	0	0	0
2	0	0	0

(a) 3-valued m unary inverter operation

$\overset{M}{\vec{x}}$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

(b) 3-valued m unary clockwise cycling operation

A number of important properties exhibited by the selected basic set provide a means for algebraic manipulation.

Theorem 3.2.3

(a) $(x_1 \cdot x_2 \cdot \ \ldots \ \cdot x_n)^K = x_1^K + x_2^K + \ldots + x_n^K$ (3.2.11)

(b) $(x_1 + x_2 + \ \ldots \ + x_n)^K = x_1^K \cdot x_2^K \cdot \ldots \cdot x_n^K$ (3.2.12)

Theorem 3.2.4

(a) $(x \cdot x^k) = 0$ (3.2.13)

(b) $(x \cdot x^K)^M = M_{m-1}$ (3.2.14)

(c) $x \cdot \vec{x} \cdot \ldots \cdot \vec{x} = 0$ (3.2.15)

where $1 \leq K \leq m-1$ and $1 \leq M \leq m-1$.

Many other properties of this algebra were given by Vranesic et al [11].

Post [8] showed that the cycling operation and the product operation are a functionally complete set, and so this expanded collection of operations is also functionally complete. Any n-variable m-valued switching function $f(x_1,...,x_n)$ has a sum-of-products canonical form:

$$f(x_1,...,x_n) = \sum_{K=1}^{m-1} \left[\prod_{v|f(v) = k} \left(\sum_{i=1}^{n} \right) \vec{x}_i^{\,m-v_1-1} \right]^K \qquad (3.2.16)$$

where $v|f(v) = k$ is the set of vertices $V = v_1,...,v_n$ for which $f(V) = k$.

The sum, product and inverter operations can be implemented using ordinary electronic gates with minor modifications; the cycling operation can be implemented by a universal cycling gate which has been designed by Vranesic, et al. [11]. The electronic implementation of the cycling gate [11] and its truth table are shown in Figure 3.2.4 and Table 3.2.4 respectively. Figure 3.2.5 shows seven basic gates of this multi-valued switching system, where k-NAND, K-NOR, and K-M gates are defined by:

(1) K-NAND gate

$z = K$ if $\min(x,y) = 0$

$= 0$ otherwise

(2) K-NOR gate

$z = K$ if $\max(x,y) = 0$

$= 0$ otherwise

(3) K-M gate

$$z = K \text{ if } \overset{M}{\vec{x}} = 0$$

$$= 0 \quad \text{otherwise}$$

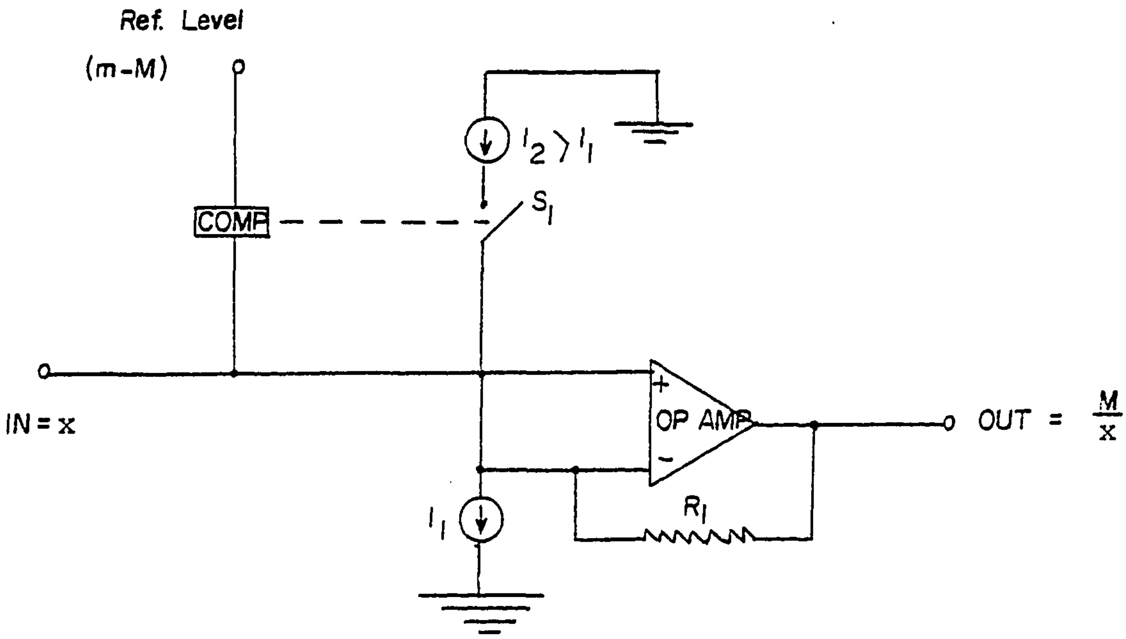


Figure 3.2.4: Cycling gate

TABLE 3.2.4

TRUTH TABLE OF THE CYCLING GATE

x	0	1 . . . m−M . . . m−2	m−1
$\overset{M}{\vec{x}}$	M	M+1. . . 0 . . . M−2	M−1

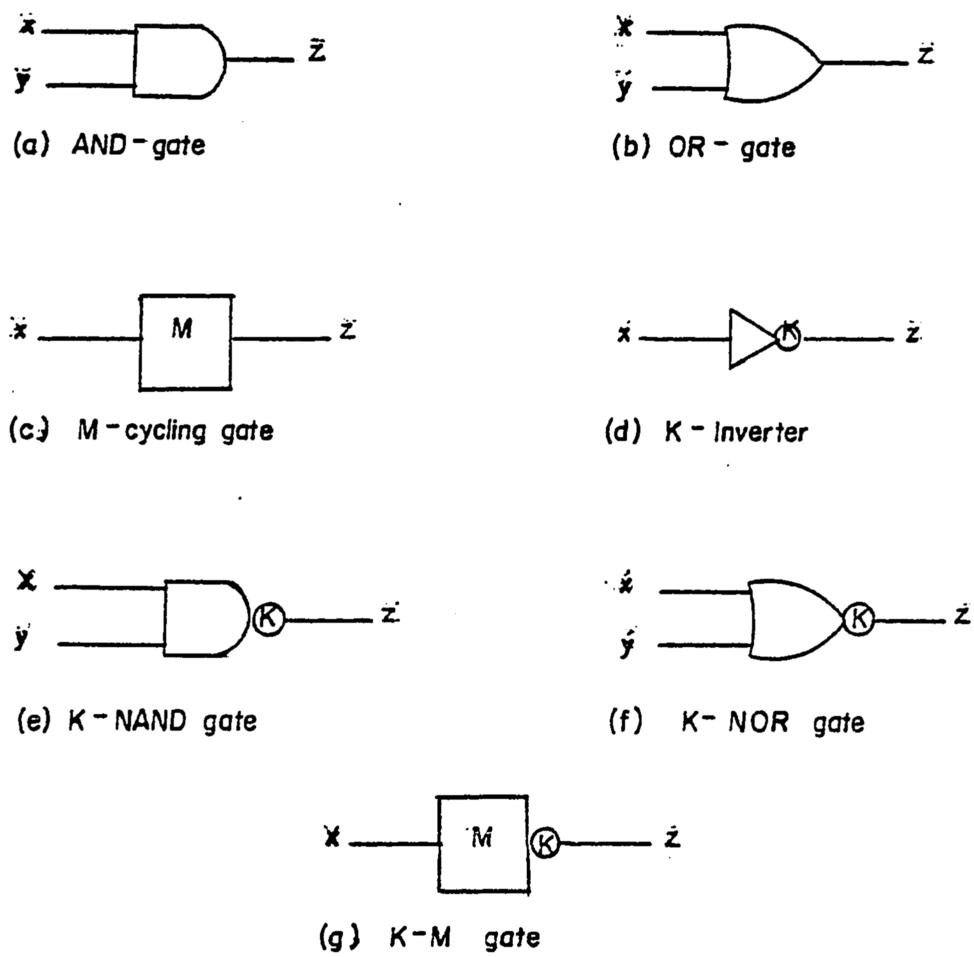


Figure 3.2.5: Seven basic gates of the multi-valued switching system derived from algebra B.

Example 3.2.3 Write the sum-of-products canonical form of the 5-valued switching function described by the truth table of Table 3.2.2.

$$f(x,y) = [(\overset{1}{\vec{x}} + y)(\overset{1}{\vec{x}} + \overset{4}{\vec{y}})(\overset{1}{\vec{x}} + \overset{3}{\vec{y}})(\overset{1}{\vec{x}} + \overset{2}{\vec{y}})(\overset{1}{\vec{x}} + \overset{1}{\vec{y}})]^{1}$$

$$+ [(x + \overset{3}{\vec{y}})(x + \overset{2}{\vec{y}})(\overset{4}{\vec{x}} + \overset{2}{\vec{y}})(\overset{4}{\vec{x}} + \overset{3}{\vec{y}})]^{2}$$

$$+ [(x + y)(x + \overset{1}{\vec{y}})]^{3} \qquad (3.2.17)$$

By applying Theorem 3.2.2 and Theorem 3.2.3, the function of Equation (3.2.17) can be minimized as follows:

$$f(x,y) = [(\overset{1}{\vec{x}} + y)(\overset{1}{\vec{x}} + \overset{4}{\vec{y}})(\overset{1}{\vec{x}} + \overset{3}{\vec{y}})(\overset{1}{\vec{x}} + \overset{2}{\vec{y}})(\overset{1}{\vec{x}} + \overset{1}{\vec{y}})]^{1} + [(x + \overset{3}{\vec{y}})(x + \overset{2}{\vec{y}})$$

$$(\overset{4}{\vec{x}} + \overset{2}{\vec{y}})(\overset{4}{\vec{x}} + \overset{3}{\vec{y}})]^{2} + [(x + y)(x + \overset{1}{\vec{y}})]^{3}$$

$$= (\overset{1}{\vec{x}} + y)^{1} + (\overset{1}{\vec{x}} + \overset{4}{\vec{y}})^{1} + (\overset{1}{\vec{x}} + \overset{3}{\vec{y}})^{1} + (\overset{1}{\vec{x}} + \overset{2}{\vec{y}})^{1} + (\overset{1}{\vec{x}} + \overset{1}{\vec{y}})^{1}$$

$$+ (x + \overset{3}{\vec{y}})^{2} + (x + \overset{2}{\vec{y}})^{2} + (\overset{4}{\vec{x}} + \overset{2}{\vec{y}})^{2} + (\overset{4}{\vec{x}} + \overset{3}{\vec{y}})^{2} + (x + y)^{3}$$

$$+ (x + \overset{1}{\vec{y}})^{3} = \overset{1}{\vec{x}}^{1}y^{1} + \overset{1}{\vec{x}}^{1}\overset{4}{\vec{y}}^{1} + \overset{1}{\vec{x}}^{1}\overset{2}{\vec{y}}^{1} + \overset{1}{\vec{x}}^{1}\overset{2}{\vec{y}}^{1} + \overset{1}{\vec{x}}^{1}\overset{1}{\vec{y}}^{1} + x^{2}\overset{3}{\vec{y}}^{2}$$

$$+ x^{2}\overset{2}{\vec{y}}^{2} + \overset{4}{\vec{x}}^{2}\overset{2}{\vec{y}}^{2} + \overset{4}{\vec{x}}^{2}\overset{3}{\vec{y}}^{2} + x^{3}y^{3} + x^{3}\overset{1}{\vec{y}}^{3}$$

$$= \overset{1}{\vec{x}}^{1}(y^{1} + \overset{1}{\vec{y}}^{1} + \overset{2}{\vec{y}}^{1} + \overset{3}{\vec{y}}^{1} + \overset{4}{\vec{y}}^{1}) + x^{2}\overset{2}{\vec{y}}^{2} + x^{2}\overset{3}{\vec{y}}^{2} + \overset{4}{\vec{x}}^{2}\overset{2}{\vec{y}}^{2} + \overset{4}{\vec{x}}^{2}\overset{3}{\vec{y}}^{2}$$

$$+ x^{3}y^{3} + x^{3}\overset{1}{\vec{y}}^{3}$$

$$= \overset{1}{\vec{x}}{}^{1} + x^{2}\overset{2}{\vec{y}}{}^{2} + x^{2}\overset{3}{\vec{y}}{}^{2} + \overset{4}{\vec{x}}{}^{2}\overset{2}{\vec{y}}{}^{2} + \overset{4}{\vec{x}}{}^{2}\overset{3}{\vec{y}}{}^{2} + x^{3}y^{3} + x^{3}\overset{1}{\vec{y}}{}^{3}$$

(3.2.18)

<u>Example 3.2.4</u> Find AND-OR realization of Equation (3.2.18).

For convenience Equation (3.2.18) is repeated below. The AND-OR realization of Equation (3.2.19) is shown in Figure 3.2.6.

$$f(x,y) = \overset{1}{\vec{x}}{}^{1} + x^{2}\overset{2}{\vec{y}}{}^{2} + x^{2}\overset{3}{\vec{y}}{}^{2} + \overset{4}{\vec{x}}{}^{2}\overset{2}{\vec{y}}{}^{2} + \overset{4}{\vec{x}}{}^{2}\overset{3}{\vec{y}}{}^{2} + x^{3}y^{3} + x^{3}\overset{1}{\vec{y}}{}^{3}$$

(3.2.19)

3.3 Relationship Between Algebra A and Algebra B

It is observed that algebras A and B may be transformed from one to the other by utilizing the following relations between the inverter and cycling operations of algebra B and the interval variable and the interval operations of algebra A: Let $1 \leq K \leq m-1$ and $1 \leq M \leq m-1$,

(1) $K\cdot \overset{m-M,m-M}{x} = \overset{M}{\vec{x}}{}^{K}$ (3.3.1)

(2) $\overset{m-M,m-M}{x} = \overset{M}{\vec{x}}\ m-1$ (3.3.2)

(3) $K\cdot \overline{\overset{m-M,m-M}{x}} = (\overset{M}{\vec{x}}{}^{K})^{K}$ (3.3.3)

(4) $(\overset{M_1}{\vec{x}_1} + \overset{M_2}{\vec{x}_2} + \ldots + \overset{M_2}{\vec{x}_n})^{K} = K\cdot(\overset{m-M_1,m-M_1}{x_1} \cdot \overset{m-M_2,m-M_2}{x_2}$

$\cdot\ldots\cdot \overset{m-M_n,m-M_n}{x_n})$ (3.3.4)

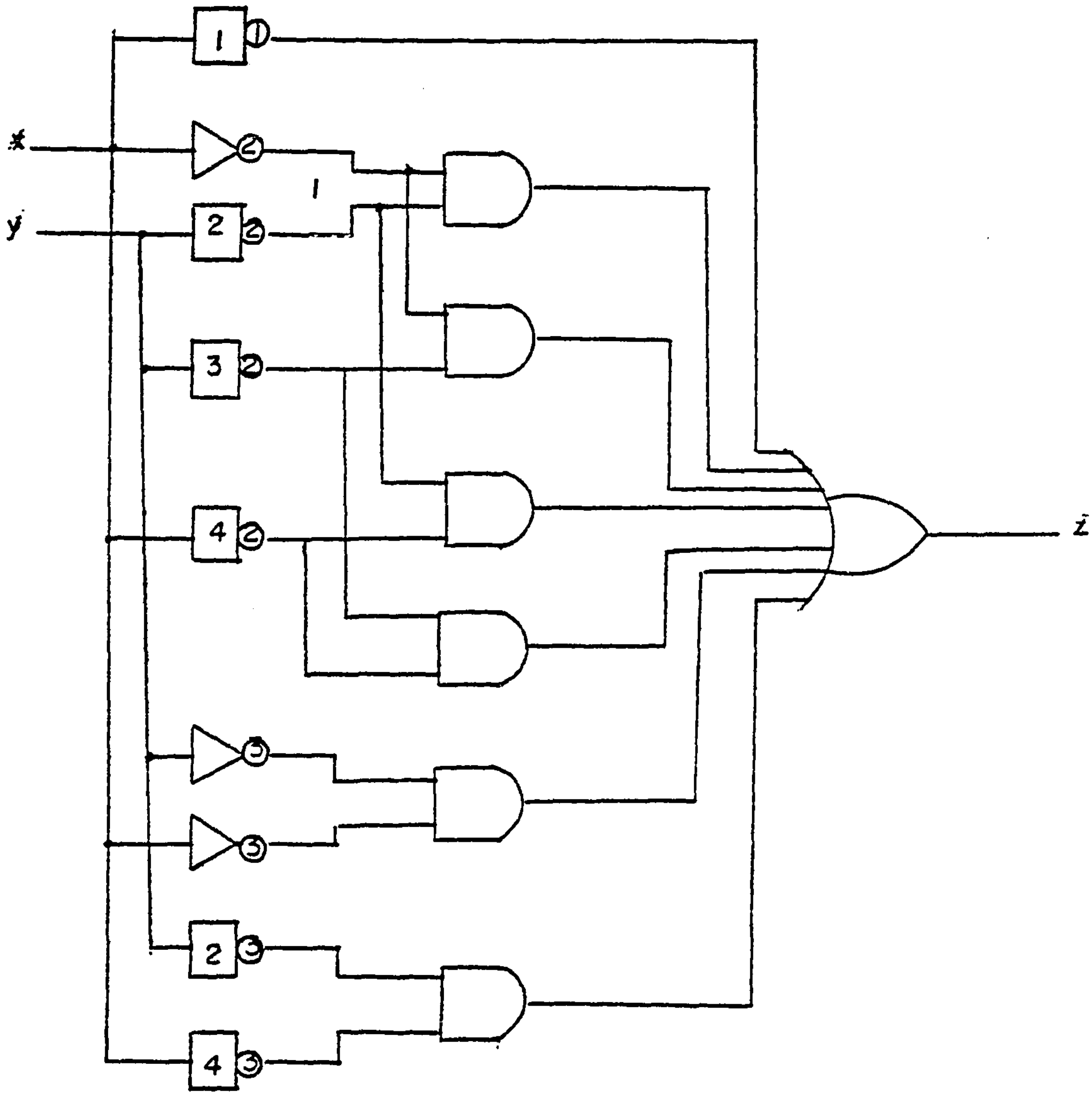


Figure 3.2.6: AND-OR Realization of Equation (3.2.19).

$$(5)\quad K\cdot \overset{a,b}{x} = {\overset{m-a}{\vec{x}}}^{K} + {\overset{m-(a+1)}{\vec{x}}}^{K} + \dots + {\overset{m-b}{\vec{x}}}^{K} \qquad 0\le a\le b\le m-1$$

(3.3.5)

For example, the function of Equation (3.2.10) of algebra A can be transformed to the function of Equation (3.2.18) of algebra B by using relation (1) of Equation (3.3.1) as follows:

$$f(x,y) = 1\cdot \overset{4,4}{x} + 2\cdot \overset{0,0}{x}\,\overset{2,2}{y} + 2\cdot \overset{0,0}{x}\,\overset{3,3}{y} + 2\cdot \overset{1,1}{x}\,\overset{2,2}{y}$$
$$+ 2\cdot \overset{1,1}{x}\,\overset{3,3}{y} + 3\,\overset{0,0}{x}\,\overset{0,0}{y} + 3\,\overset{0,0}{x}\,\overset{4,4}{y}$$

or

$$f(x,y) = {\overset{1}{\vec{x}}}^{1} + x^{2}{\overset{3}{\vec{y}}}^{2} + x^{2}{\overset{2}{\vec{y}}}^{2} + {\overset{4}{\vec{x}}}^{2}{\overset{3}{\vec{y}}}^{2} + {\overset{4}{\vec{x}}}^{2}{\overset{2}{\vec{y}}}^{2} + x^{3}y^{3} + x^{3}{\overset{1}{\vec{y}}}^{3}$$

Algebras A and B are found to be equivalent to each other because each algebra can be transformed from one to the other by using the above relations of Equations (3.3.1) through (3.3.5). Algebra A is more commonly used in literature because of the following two facts: (1) any variable or function of algebra A can be complemented in a similar way as in Boolean switching algebra and (2) any function of algebra A can be minimized using either, k-valued map or the tabular method which are also similar to the minimization techniques being used in Boolean switching algebra. These two advantages of algebra A are not directly obvious in algebra B because the complement of a variable or a function of algebra B is not defined. The minimization technique of any function of algebra B must use Theorems (3.2.2) and (3.2.3) which is not practical for large number of input variables.

Algebra A will be used through the remaining of the dissertation. But all equations, expressions, formulas and relations of algebra A can be easily transformed into algebra B by using Equations (3.3.1) through (3.3.5) because Algebras A and B are one-to-one correspondence.

3.4 Tabular Minimization Technique

Multi-valued map is a very powerful design tool, but it does have certain drawbacks. First, it is a trial and error method which does not offer any guarantee of producing the best result. Second, for functions of 5 or more variables, it is difficult to select the smallest possible set of products from a multi-valued map. It is found that Quine [21] and McCluskey [22] tabular method for two-valued case can be extended to multi-valued. Multi-valued tabular method corrects these deficiencies and it is suitable for computer programming.

The multi-valued tabular minimization procedure is different from Quine and McCluskey method. The procedure is illustrated by the following example.

Example 3.4.1

Minimized the following 5-valued function:

$$f(x_1,x_2x_3) = \Sigma[1\cdot(1,2,3,7,8,12,13,17,18,22,23,70,91,92,114,116,124) + 2\cdot(70,71,75,76,80,81,85,86,88,114,124) + 3\cdot(104,109,114,119,124) + 4\cdot(95,99)]$$

Step 1: Delete all minterms that appear in the function more than one except the minterms associated with the highest value of K. For example, the minterms 114 and 124 appear in 1-summation, 2-summation, and 3-summation where 1,2, and 3 are the values of K. Minterms 114 and 124 must appear only in 3-summation. Thus, the above function is written as follows:

$$f(x_1,x_2,x_3) = \Sigma\ [1\cdot(1,2,3,7,8,12,13,17,18,22,23,91,92,116) + 2\cdot(70,71,75,76,80,81,85,86,88) + 3\cdot(104,109,114,119,124) + 4\cdot(95,99)]$$

Step 2: Find corresponding m-valued numbers of given minterms, 5-valued numbers of minterms of the above function is listed in the following table.

Decimal Number	5-Valued Number	Decimal Number	5-Valued Number
1	001	81	311
2	002	85	320
3	003	86	321
7	012	88	323
8	013	91	331
12	022	92	332
13	023	95	340
17	032	99	344
18	033	104	404
22	042	109	414
23	043	114	424
70	240	116	431
71	241	110	434
75	300	124	444
76	301		
80	310		

Step 3: Construct one implicant table for each K-minterms summation. Implicant table is constructed as follows:

(i) Write all minterms of K-summation in the first column of the table.

(ii) Combine any two minterms of first column as 1-cube if they have the same numbers in all positions except one position, provided that the differences between the corresponding numbers in this position are equal to one.

(iii) Put check mark ✓ for those minterms (cubes) that are combined.

(iv) Delete all those cubes that appear more than once in a column except one of them.

(v) Repeat (ii), (iii), and (iv) till no cubes can be found to be combined.

Implicant tables for 1-summation, 2-summation, 3-summation, and 4-summation are shown in Tables 3.4.1, 3.4.2, 3.4.3, and 3.4.4, respectively.

<u>Step 4</u>: Construct one implicant covering table for each implicant table to find the necessary implicants that need to be included in the minimal function. The rows of the implicant covering table contain the implicants obtained in Step 3 and its columns contain the minterms. The implicant covering table of Table 3.4.1 is given by Table 3.4.5. Implicant covering tables of Tables 3.4.2, 3.4.3, and 3.4.4 can be similarly constructed. Thus, they are omitted.

TABLE 3.4.1

IMPLICANT TABLE FOR 1-MINTERMS SUMMATION

Minterms	1-Cubes	2-Cubes	3-Cubes	4-Cubes	5-Cubes
001✓	00(12)✓	00(123)c*	0(012)(23)	0(0123)(23)	0(01234)(23)=
002✓	00(23)✓	0(01)(23)✓	0(123)(23)	0(1234)(23)	OX(23)d*
003✓	01(23)✓	0(12)(23)✓	0(234)(23)		
012✓	0(12)2✓	0(23)(23)✓			
013✓	0(12)3✓	0(34)(23)✓			
022✓	02(23)✓	~~0(12)(23)~~			
023✓	0(23)2✓	~~0(23)(23)~~			
032✓	0(23)3✓	~~0(34)(23)~~			
033✓	03(23)✓				
042✓	0(34)2✓				
043✓	0(34)3✓				
331✓	04(23)✓				
332✓	33(12)a*				
431✓	(34)31b*				

TABLE 3.4.2

IMPLICANT TABLE FOR 2-MINTERMS SUMMATION

Minterms	1-Cubes	2-Cubes	3-Cubes
240✓	24(01)f*	3(01)(01)✓	3(012)(01)g*
241✓	30(01)✓	~~3(01)(01)~~	~~3(012)(01)~~
300✓	3(01)0✓	3(12)(01)✓	
301✓	3(01)1✓	3(012)0✓	
310✓	31(01)✓	3(012)1✓	
311✓	3(12)0✓	3(12)(01)✓	
320✓	32(01)✓		
321✓	3(12)1✓		
323e*			

TABLE 3.4.3

IMPLICANT TABLE FOR 3-MINTERMS SUMMATION

Minterms	1-Cubes	2-Cubes	3-Cubes	4-Cubes
404✓	4(01)4✓	4(012)4✓	4(0123)4✓	4(01234)4=4x4 j*
414✓	4(12)4✓	4(123)4✓	4(1234)4✓	
424✓	4(23)4✓	4(234)4✓		
434✓	4(34)4✓			
444✓				

TABLE 3.4.4

IMPLICANT TABLE FOR 4-MINTERMS SUMMATION

Minterms	
340	h*
344	i*

TABLE 3.4.5

IMPLICANT COVERING TABLE OF TABLE 3.4.1

	1	2	3	7	8	12	13	17	18	22	23	91	92	116
*a												✓	✓	
*b												✓		✓
*c	✓	✓		✓										
*d		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Finally, the minimized function f is:

$$f(x_1,x_2,x_3) = 1\cdot(a+b+c+d) + 2\cdot(e+f+g) + 3\cdot(j) + 4\cdot(h+i)$$

$$= 1\cdot(\,x_1^{3,3}\;x_2^{3,3}\;x_3^{1,2} + x_1^{3,4}\;x_2^{3,3}\;x_3^{1,1} + x_1^{0,0}\;x_2^{0,0}\;x_3^{1,3} + x_1^{0,0}\;x_3^{2,3})$$

$$+\ 2\cdot(x_1^{3,3}\;x_2^{2,2}\;x_3^{3,3} + x_1^{2,2}\;x_2^{4,4}\;x_3^{0,1} + x_1^{3,3}\;x_2^{0,2}\;x_3^{0,1})$$

$$+\ 3\cdot\,x_1^{4,4}\;x_3^{4,4} + 4(\,x_1^{3,3}\;x_2^{4,4}\;x_3^{0,0} + x_1^{3,3}\;x_2^{4,4}\;x_3^{4,4})$$

CHAPTER IV

MULTI-VALUED DIGITAL CALCULUS

4.1 Introduction

As we have mentioned in Chapter 1 that Reed [1] had discovered the Boolean derivative in 1954. Two-valued digital calculus, introduced first by Akers [2] in 1959, has been more thoroughly investigated by Thayse [4] in 1971. In this chapter, we introduce the multi-valued digital calculus which is the general case that can be used to any m-valued switching function. The main purpose of this chapter is to present the multi-valued partial derivative and the multi-valued total differential.

4.2 The Partial Derivative

In this section, we introduce the partial derivative of a multi-valued switching function defined on the multi-valued switching algebra proposed by Su and Sarris [12]. First we define:

<u>Definition 4.2.1</u>

Define the EXCLUSIVE-OR (EX-OR) operation of the two multi-valued variables x and y as:

$$x \oplus y = x+y \quad \text{if } x \neq y$$
$$= 0 \quad \text{if } x = y \tag{4.2.1}$$

This EX-OR definition satisfies the commutative property $x \oplus y = y \oplus x$ and the distributive property $z(x \oplus y) = z(x+y) = zx+zy$ for $x \neq y$ and $z(x \oplus y) = zx \oplus zy = 0$ for $x=y$, but it does not satisfy the associate property $(x \oplus y) \oplus z = x \oplus (y \oplus z)$. Fortunately, this associate property is not required for fault detection and hazard detection and elimination that they will be discussed in Chapters 5, 6, and 7 respectively. EX-OR operation for 3-valued and 5-valued functions using Definition 4.2.1 are given in Tables 4.2.1a and 4.2.1b respectively.

TABLE 4.2.1

$\oplus$	0	1	2
0	0	1	2
1	1	0	2
2	2	2	0

(a) 3-Valued EX-OR Operation

$\oplus$	0	1	2	3	4
0	0	1	2	3	4
1	1	0	2	3	4
2	2	2	0	3	4
3	3	3	3	0	4
4	4	4	4	4	0

(b) 5-Valued EX-OR Operation

The partial derivative of a multi-valued switching function is defined as follows:

<u>Definition 4.2.2</u>

Let $f(x_1,\ldots,x_n, {}^{a_i,b_i}x_i)$ be a multi-valued function of n variables $x_1,x_2,\ldots,x_n$. The partial derivative is defined as:

$$\frac{\partial f(x_1,\ldots,x_n, {}^{a_i,b_i}x_i)}{\partial\, {}^{a_i,b_i}x_i} = f(x_1,\ldots,x_n, {}^{a_i,b_i}x_i) \oplus f(x_1,\ldots,x_n, {}^{\overline{a_i,b_i}}x_i) \tag{4.2.2}$$

The above definition is equivalent to the following definition.

<u>Definition 4.2.2'</u>

$$\frac{\partial f(x_1,\ldots,x_n, {}^{a_i,b_i}x_i)}{\partial\, {}^{a_i,b_i}x_i} = f(x_1,\ldots,x_n,m-1) \oplus f(x_1,\ldots,x_n,0) \tag{4.2.3}$$

<u>Example 4.2.1</u>

Find the partial derivative of the following 3-valued switching function with respect to (a) ${}^{0,1}x_2$ and (b) ${}^{1,1}x_1$.

$$f(x_1,x_2,x_1) = 1\cdot {}^{1,1}x_1\ {}^{0,1}x_2 + 2\cdot {}^{\overline{0,1}}x_2\ {}^{2,2}x_3 + 2\cdot {}^{0,0}x_1\ {}^{0,0}x_3$$

(a) After using Definition 4.2.2', the partial derivative of f with respect to ${}^{0,1}x_2$ becomes:

$$\frac{\partial f}{\partial \overset{0,1}{x_2}} = (1\cdot \overset{1,1}{x_1} \cdot(2) + 2\cdot(0) \overset{2,2}{x_3} + 2 \overset{0,0}{x_1} \overset{0,0}{x_3}) \oplus (1\cdot \overset{1,1}{x_1}(0) + 2\cdot (2) \overset{2,2}{x_3} + 2\cdot \overset{0,0}{x_1} \overset{0,0}{x_3})$$

$$= (1\cdot \overset{1,1}{x_1} + 2\cdot \overset{0,0}{x_1} \overset{0,0}{x_3}) \oplus (2\cdot \overset{2,2}{x_3} + 2\cdot \overset{0,0}{x_1} \overset{0,0}{x_3})$$

$\partial f/\partial \overset{0,1}{x_2}$ can be simplified by using the 3-valued map as follows:

$\frac{\partial f}{\partial \overset{0,1}{x_2}} =$

$x_3 \backslash x_1$	0	1	2
0	2	1	
1		1	
2		1	

$\oplus$

$x_3 \backslash x_1$	0	1	2
0	2		
1			
2	2	2	2

$=$

$x_3 \backslash x_1$	0	1	2
0		1	
1		1	
2	2	2	2

$$\frac{\partial f}{\partial \overset{0,1}{x_2}} = 1\cdot \overset{1,1}{x_1} + 2\cdot \overset{2,2}{x_3}$$

(b) The partial derivative of f with respect to $\overset{1,1}{x_1}$ is given by:

$$\frac{\partial f}{\partial \overset{1,1}{x_1}} = (1\cdot(2) \overset{0,1}{x_2} + 2 \overset{\overline{0,1}}{x_2} \overset{2,2}{x_3} + 2 \overset{0,0}{x_1} \overset{0,0}{x_3}) \oplus (1\cdot(0) \overset{0,1}{x_2} + 2\cdot \overset{\overline{0,1}}{x_2} \overset{2,2}{x_3} + 2\cdot \overset{0,0}{x_1} \overset{0,0}{x_3})$$

$$= (1\cdot \overset{0,1}{x_2} + 2 \overset{\overline{0,1}}{x_2} \overset{2,2}{x_3} + 2 \overset{0,0}{x_1} \overset{0,0}{x_3}) \oplus (2\cdot \overset{\overline{0,1}}{x_2} \overset{2,2}{x_3} + 2\cdot \overset{0,0}{x_1} \overset{0,0}{x_3})$$

$x_3 \backslash x_1x_2$	00	01	02	10	11	12	20	21	22
0	2	2	2	1	1		1	1	
1	1	1		1	1		1	1	
2	1	1	2	1	1	2	1	1	2

$\oplus$

$x_3 \backslash x_1x_2$	00	01	02	10	11	12	20	21	22
0	2	2	2						
1									
2			2			2			2

=

$x_3 \backslash x_1x_2$	00	01	02	10	11	12	20	21	22
0				1	1		1	1	
1	1	1		1	1		1	1	
2	1	1		1	1		1	1	

$$\frac{\partial f}{\partial x_1^{1,1}} = 1 \cdot x_2^{0,1} x_3^{1,2} + 1 \cdot x_1^{1,2} x_2^{0,1}$$

The multiple partial derivative is defined as follows:

<u>Definition 4.2.3</u>

$$\frac{\partial f}{\partial x_i^{a_i,b_i} x_{i+1}^{a_{i+1},b_{i+1}} \cdots x_{i+\ell}^{a_{i+\ell},b_{i+\ell}}} = f(x_1,\ldots,x_n, x_i^{a_i,b_i}, x_{i+1}^{a_{i+\ell},b_{i+1}}, \ldots, x_{i+\ell}^{a_{i+\ell},b_{i+\ell}})$$

$$\oplus\ f(x_1,\ldots,x_n,\ x_i^{\overline{a_i,b_i}},\ x_{i+1}^{\overline{a_{i+1},b_{i+1}}},\ \ldots,\ x_{i+\ell}^{\overline{a_{i+\ell},b_{i+\ell}}}) \tag{4.2.4}$$

<u>Example 4.2.2</u>

Find the partial derivative of the following 5-valued switching function with respect to $x_1^{1,1}\ x_3^{3,4}$.

$$f(x_1,x_2,x_3) = 1\cdot x_1^{1,1}\ x_2^{2,3} + 2\cdot x_1^{4,4}\ x_3^{3,4} + 4\cdot x_2^{0,0}\ x_3^{\overline{3,4}}$$

$$\frac{\partial f}{\partial(x_1^{1,1}\ x_3^{3,4})} = f(x_1,x_2,x_3,\ x_1^{1,1},\ x_3^{3,4}) \oplus f(x_1,x_2,x_3,\ x_1^{\overline{1,1}},\ x_3^{\overline{3,4}})$$

$$= (1\cdot x_1^{1,1}\ x_2^{2,3} + 2\cdot x_1^{4,4}\ x_3^{3,4} + 4\cdot x_2^{0,0}\ x_3^{\overline{3,4}})$$

$$\oplus\ (1\cdot x_1^{\overline{1,1}}\ x_2^{2,3} + 2\cdot x_1^{4,4}\ x_3^{\overline{3,4}} + 4\cdot x_2^{0,0}\ x_3^{3,4})$$

=

x_1x_1 / x_3	00	01	02	03	04	10	11	12	13	14	20	21	22	23	24	30	31	32	33	34	40	41	42	43	44
0	4					4		1	1		4					4					4				
1	4					4		1	1		4					4					4				
2	4					4		1	1		4					4					4				
3								1	1												2	2	2	2	2
4								1	1												2	2	2	2	2

⊕

x_1x_2 / x_3	00	01	02	03	04	10	11	12	13	14	20	21	22	23	24	30	31	32	33	34	40	41	42	43	44
0			1	1									1	1				1	1		2	2	2	2	2
1			1	1									1	1				1	1		2	2	2	2	2
2			1	1									1	1				1	1		2	2	2	2	2
3	4		1	1		4					4		1	1		4		1	1		4		1	1	
4	4		1	1		4					4		1	1		4		1	1		4		1	1	

=

x_1x_2 / x_3	00	01	02	03	04	10	11	12	13	14	20	21	22	23	24	30	31	32	33	34	40	41	42	43	44
0	4		1	1		4		1	1		4		1	1		4		1	1		4	2	2	2	2
1	4		1	1		4		1	1		4		1	1		4		1	1		4	2	2	2	2
2	4		1	1		4		1	1		4		1	1		4		1	1		4	2	2	2	2
3	4		1	1		4		1	1		4		1	1		4		1	1		4	2	2	2	2
4	4		1	1		4		1	1		4		1	1		4		1	1		4	2	2	2	2

Therefore,

$$\frac{\partial f}{\partial(x_1^{1,1}\ x_3^{3,4})} = 1\cdot x_1^{\overline{4,4}}\ x_2^{2,3} + 2\cdot x_1^{4,4}\ x_2^{\overline{0,0}} + 4\cdot x_2^{0,0}$$

4.3 The Total Differential

In this section, the total differential of a multi-valued switching function is introduced. The total differential of a multi-valued switching function is defined in the following definition which is similar to the ordinary total differential.

Definition 4.3.1

The total differential of a multi-valued switching function $f(x_1,\ldots,x_n, x_i^{a_{ij},b_{ij}})$ is defined as follows:

$$df = \sum_{i=1}^{n} \sum_{j} \frac{\partial f}{\partial x_i^{a_{ij},b_{ij}}} \, d\, x^{a_{ij},b_{ij}} \qquad (4.3.1)$$

where Σ is the summation over the OR operation and,

$$d\, x_i^{a_{ij},b_{ij}} = m-1 \qquad \text{if } x_i^{a_{ij},b_{ij}} \text{ changes}$$

$$= 0 \qquad \text{if } x_i^{a_{ij},b_{ij}} \text{ does not change}$$

$d\, x_i^{a_{ij},b_{ij}}$ is a two-valued vairable that its largest value is equal to m-1 and its smallest value is equal to zero.

Example 4.3.1

Find the total differential of the following 3-valued switching function.

$$f(x_1,x_2) = 1\cdot \overset{1,1}{x_1}\ \overset{1,2}{x_2} + 2\cdot \overset{0,0}{x_1}$$

$$df = \frac{\partial f}{\partial \overset{1,1}{x_1}}\, d\,\overset{1,1}{x_1} + \frac{\partial f}{\partial \overset{0,0}{x_1}}\, d\,\overset{0,0}{x_1} + \frac{\partial f}{\partial \overset{1,2}{x_2}}\, d\,\overset{1,2}{x_1}$$

$$\frac{\partial f}{\partial \overset{1,1}{x_1}} = (1\cdot \overset{1,2}{x_2} + 2\ \overset{0,0}{x_1}) \oplus 2\cdot \overset{0,0}{x_1} = 1\cdot \overset{1,2}{x_1}\ \overset{1,2}{x_2}$$

$$\frac{\partial f}{\partial \overset{0,0}{x_1}} = (1\cdot \overset{1,1}{x_1}\ \overset{1,2}{x_2} + 2) \oplus 1\cdot \overset{1,1}{x_1}\ \overset{1,2}{x_2} = 2$$

$$\frac{\partial f}{\partial \overset{1,2}{x_2}} = (1\cdot \overset{1,1}{x_1} + 2\cdot \overset{0,0}{x_1}) \oplus 2\cdot \overset{0,0}{x_1} = 1\cdot \overset{1,1}{x_1}$$

Therefore,

$$df = 1\cdot \overset{1,2}{x_1}\ \overset{1,2}{x_2}\, d\,\overset{1,1}{x_1} + 2\cdot d\,\overset{0,0}{x_1} + 1\cdot \overset{1,1}{x_1}\, d\,\overset{1,2}{x_2}$$

4.4 Computer Method For Computing Partial Derivative

In the foregoing sections, we have discussed multi-valued partial derivatives and differentials of multi-valued switching functions. When the number of variables and the value of the function increase, the map method becomes more and more laborious and it will involve more work. In this section, a computer method for computing $\partial f/\partial \, x_i^{a_i,b_i}$ is presented. This method is very convenient for computer implementation to compute the multi-valued partial derivative and the multi-valued total differential.

Example 4.4.1

Consider the following 3-variables, 5-valued function

$$f(x_1,x_2,x_3) = 2\cdot[45,46,49,70,71,74,120,121,124]$$
$$+ 3\cdot[77,78,79,82,83,84,87,88,89,92,93,94,97,98,99]$$
$$+ 4\cdot[10,11,12,13,14,15,16,17,18,19,20,21,22,23,24]$$

Find the partial derivative with respect to $x_1^{3,3}$.

Step 1: Find 5-valued representations of the above minterms. They are shown in the first and second columns of Table 4.4.1.

Step 2: Create new minterms by changing $x_1^{3,3}$ of the minterms to $x_1^{\overline{3,3}}$ and delete those (the original and the minterm generated by it) which appear twice. They are shown in the third, fourth, and fifth columns of Table 4.4.1.

Step 3: Find the minimal expression of $\partial f/\partial \, x_1^{3,3}$ using the tabular minimization technique described in Section 3.4. The minimized $\partial f/\partial \, x_1^{3,3}$ is found to be:

$$\frac{\partial f}{\partial \, x_1^{3,3}} = 3\cdot x_1^{1,4} \, x_3^{2,4} + 3\cdot x_1^{0,0} \, x_2^{0,1} \, x_3^{2,4}$$

TABLE 4.4.1

$x_1x_2x_3$ k	$x_1x_2x_3$ k	$x_1x_2x_3$ k	$x_1x_2x_3$ k	$x_1x_2x_3$ k
~~140(2)~~	~~030(4)~~	~~140(2)~~	014(3)	142(3)
144(2)	~~031(4)~~	~~141(2)~~	114(3)	242(3)
~~240(2)~~	~~032(4)~~	~~144(2)~~	214(3)	442(3)
~~241(2)~~	~~033(4)~~	~~240(2)~~	414(3)	~~043(3)~~
~~244(2)~~	~~034(4)~~	~~241(2)~~	~~022(3)~~	143(3)
~~440(2)~~	~~040(4)~~	~~244(2)~~	122(3)	243(3)
~~441(2)~~	~~041(4)~~	~~440(2)~~	222(3)	443(3)
~~444(2)~~	~~042(4)~~	~~441(2)~~	422(3)	~~044(3)~~
302(3)	~~043(4)~~	~~444(2)~~	~~023(3)~~	144(3)
303(3)	~~044(4)~~	002(3)	123(3)	244(3)
304(3)		102(3)	223(3)	444(3)
312(3)		202(3)	423(3)	~~020(4)~~
313(3)		402(3)	~~024(3)~~	~~021(4)~~
314(3)		002(3)	124(3)	~~022(4)~~
322(3)		102(3)	224(3)	~~023(4)~~
323(3)		202(3)	424(3)	~~024(4)~~
324(3)		402(3)	~~032(3)~~	~~030(4)~~
332(3)		004(3)	132(3)	~~031(4)~~
333(3)		104(3)	232(3)	~~032(4)~~
334(3)		204(3)	432(3)	~~033(4)~~
342(3)		404(3)	~~033(3)~~	~~034(4)~~
343(3)		012(3)	133(3)	~~040(4)~~
344(3)		112(3)	233(3)	~~041(4)~~
~~020(4)~~		212(3)	433(3)	~~042(4)~~
~~021(4)~~		412(3)	~~034(3)~~	~~043(4)~~
~~022(4)~~		013(3)	134(3)	~~044(4)~~
~~023(4)~~		113(3)	234(3)	
~~024(4)~~		213(3)	434(3)	
		413(3)	~~042(3)~~	

(a) Minterms of the function of example 4.4.1

(b) Minterms created by changing $x_1^{3,3}$ of (a) to $x_1^{\overline{3,3}}$ for computing $\partial f/\partial x_1^{3,3}$.

CHAPTER V

FAULT DETECTION IN COMBINATIONAL CIRCUITS

5.1 Introduction

Digital systems are built with more and more complexity. The fault testing and diagnosis of digital circuits becomes an important part of digital system design. After a digital circuit is designed, built and properly constructed, it may be disabled by an error. The process of applying tests and determining whether a digital circuit is fault free or not is generally known as fault detection. A number of methods [18, Chapter 7, 23-27] have already been developed for fault detection in binary circuits. One of these methods is the Boolean difference method [3] which receives our special attention, since it is conceptually simple and straightforward for deriving a set of tests to detect the faults in combinational binary circuits. Fault detection in combinational binary circuits was presented in section 2.7.

Just like combinational binary circuits, multi-valued combinational switching circuits may be disabled by an error [3]. Thus, in this chapter, the Boolean difference method for detecting faults in binary circuits is generalized to multi-valued switching systems. The derivation of a complete set of tests (the set of all possible tests) for detecting any single fault in any multi-valued combinational switching circuit is presented.

5.2 Single Faults Detection on Input Lines

In this section, only those single faults that exist on the input line of the circuit are considered. A circuit becomes faulty when the value of its output is different from the corresponding value of its truth table at any input state. The faults considered here are assumed to be fixed or permanent or nontransient faults, i.e., without having them fixed or repaired, the faults will be permanently there. Such faults are those which cause a wire to be stuck at -0 (s-a-0) or stuck at -k(s-a-k), where k=1,...,m-1. There are some faults in a circuit that are undetectable, thus we define:

Definition 5.2.1

A fault in a multi-valued combinational switching circuit is said to be undetectable if the faulty and normal values of the output of the circuit are the same, otherwise the fault is detectable.

Two indistinguishable faults are defined in the following definition.

Definition 5.2.2

Two faults are said to be indistinguishable if their output faulty functions are equal.

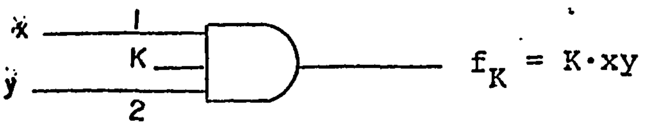
Theorem 5.2.1

Stuck-at-k (s-a-k) faults that exist at the inputs of K-AND gate or K-OR gate are indistinguishable for all $K \leq k \leq m-1$ where $k = 1,2, \ldots, m-1$.

The proof of this theorem is obvious and may thus be omitted.

Example 5.2.1

Find all undetectable and indistinguishable faults that exist on line 1 of 3-valued K-AND gate shown below for $K = 1$ and 2.



(a) $K = 1$

$f_{10} = 1 \cdot xy$

TABLE 5.2.1

Normal, s-a-0, s-a-1 and s-a-2 Output Functions for K = 1

x	y	f_{10}	f_{110}	f_{111}	f_{112}
0	0	0	0	0	0
0	1	0	0	1	1
0	2	0	0	1	1
1	0	0	0	0	0
1	1	1	0	1	1
1	2	1	0	1	1
2	0	0	0	0	0
2	1	1	0	1	1
2	2	1	0	1	1

The functions f_{10}, f_{110}, f_{111} and f_{112} that are shown in Table 5.2.1 (for K = 1) correspond to normal, s-a-0, s-a-1 and s-a-2 output functions, respectively. There are no undetectable faults because $f_{110} \neq f_{10}$, $f_{111} \neq f_{10}$ and $f_{112} \neq f_{10}$ while f_{111} and f_{112} are two indistinguishable faults because $f_{111} = f_{112}$.

(b) $K = 2$

$f_{20} = 2 \cdot xy$

TABLE 5.2.2

Normal, s-a-0, s-a-1 and s-a-2 Output Functions for K = 2

x	y	f_{20}	f_{210}	f_{211}	f_{212}
0	0	0	0	0	0
0	1	0	0	1	1
0	2	0	0	1	2
1	0	0	0	0	0
1	1	1	0	1	1
1	2	1	0	1	2
2	0	0	0	0	0
2	1	1	0	1	1
2	2	2	0	1	2

From Table 5.2.2 for K = 2, all faults are detectable because $f_{210} \neq f_{20}$, $f_{211} \neq f_{20}$, and $f_{212} \neq f_{20}$. Also, all faults are distinguishable because $K = m-1 = 2$.

The input variable ${}^{a_i,b_i}x_i$ may be used more than once at the input of the circuit such that the multi-valued constant K associated with the variable ${}^{a_i,b_i}x_i$ may take more than one value. When this situation occurs, the faults on each line connected to ${}^{a_i,b_i}x_i$ need to be detected seperately. Multi-valued partial derivative defined in Definitions 4.2.2 and 4.2.2$'$ need to be redefined such that this problem can be solved. It is found that this problem can be solved if multi-valued partial derivative is defined with respect to $K\,{}^{a_i,b_i}x_i$ as follows:

<u>Definition 5.2.3</u>

$$\frac{\partial f(x_1,\ldots,x_n,{}^{a_i,b_i}x_i)}{\partial(K\,{}^{a_i,b_i}x_i)} = f(x_1,\ldots,x_n,{}^{a_i,b_i}x_i) \oplus f(x_1,\ldots,x_n,\overline{{}^{a_i,b_i}x_i}) \tag{5.2.1}$$

<u>Definition 5.2.4</u>

$$\frac{\partial f(x_1,\ldots,x_n,{}^{a_i,b_i}x_i)}{\partial(K\,{}^{a_i,b_i}x_i)} = f(x_1,\ldots,x_n,m-1) \oplus f(x_1,\ldots,x_n,0) \tag{5.2.2}$$

Definitions 5.2.3 and 5.2.4 will be used only in this chapter for the purpose of fault detection; otherwise Definitions 4.2.2 and 4.2.2$'$ are to be used elsewhere.

The following theorem shows that the set of tests for detecting any single fault on any input line i of a multi-valued combination circuit can be obtained by using multi-valued partial derivative defined in Definitions 5.2.3 and 5.2.4.

Theorem 5.2.2

Let $f(x_1,\ldots,x_n, {}^{a_i,b_i}x_i)$ be the output function of a multi-valued combinational circuit. $(a_1\ldots a_n)$ is a test of s-a-0 or s-a-k fault on the input line i if and only if

$$\left.\frac{\partial f}{\partial\left(K\,{}^{a_i,b_i}x_i\right)}\right|_{a_1\ldots a_n} \leq L \qquad 0 < L \leq m-1 \qquad (5.2.3)$$

where $a_1\ldots a_n \ \varepsilon\,(0,1,\ldots,m-1)$

and K is a multi-valued constant associated with the gate of the input variable ${}^{a_i,b_i}x_i$

Proof:

From the definition of multi-valued partial derivative, we have

$$\frac{\partial f}{\partial\left(K\,{}^{a_i,b_i}x_i\right)} = f(x_1,\ldots,x_n,m-1) \oplus f(x_1,\ldots,x_n,0)$$

when $\left.\frac{\partial f}{\partial\left(K\,{}^{a_i,b_i}x_i\right)}\right|_{a_1\ldots a_n} \leq L,\ f(x_1,\ldots,x_n,m-1)$

$\neq f(x_1,\ldots,x_n,0)$ and that when $\left.\frac{\partial f}{\partial\left(K\,{}^{a_i,b_i}x_i\right)}\right|_{a_1\ldots a_n} =0,$

$f(x_1,\ldots,x_n,m-1) = f(x_1,\ldots,x_n,0)$. Thus,

$$\left.\frac{\partial f}{\partial\left(K^{a_i,b_i}x_i\right)}\right|_{a_1\ldots a_n} \leq L \text{ is the condition for } f(x_1,\ldots x_n,$$

${}^{a_i,b_i}x_i{}^{a_i,b_i}$) being dependent on the variable ${}^{a_i,b_i}x_i{}^{a_i,b_i}$, i.e., line i is in error and $a_1\ldots a_n$ is a test to detect this error.

The following lemmas will be used for detecting single fault on any input line i (primarily line i) of a multi-valued combinational circuit.

<u>Lemma 5.2.1</u>:

Set of tests T for detecting both s-a-0 and s-a-k faults on an input line i is computed from Equation (5.2.3).

<u>Lemma 5.2.2</u>:

Set of tests T_0 for detecting s-a-0 fault on an input line i is computed from the following equation.

$$x_i^{a_i,b_i} \left.\frac{\partial f}{\partial(K^{a_i,b_i}x_i^{a_i,b_i})}\right|_{a_1\ldots a_n} \leq L \quad 0 < L \leq m-1 \qquad (5.2.4)$$

<u>Lemma 5.2.3</u>:

Set of tests T_k for detecting s-a-k faults on an input line i is computed from the following equation:

$$\overline{{}^{a_i,b_i}x_i} \quad \frac{\partial f}{\partial (K\, {}^{a_i,b_i}x_i)} \leq L \qquad 0 < L \leq m-1 \tag{5.2.5}$$

Lemma 5.2.4:

The relationship between T, T_0 and T_k is:

$$T = T_0 \cup T_k. \tag{5.2.6}$$

Example 5.2.2

Consider 3-valued function $f(x,y)$ of Equation (5.2.7) in which its AND-OR realization is shown in Figure 5.2.1. Suppose it is desired to derive sets of tests for detecting s-a-0, s-a-1 and s-a-2 faults on lines i and j indicated in the figure.

$$f(x,y) = 1 \cdot {}^{0,0}x\ {}^{0,0}y + 1 \cdot {}^{2,2}x\ {}^{0,0}y + 2 \cdot {}^{0,0}x\ {}^{2,2}y + 2 \cdot {}^{2,2}x\ {}^{2,2}y \tag{5.2.7}$$

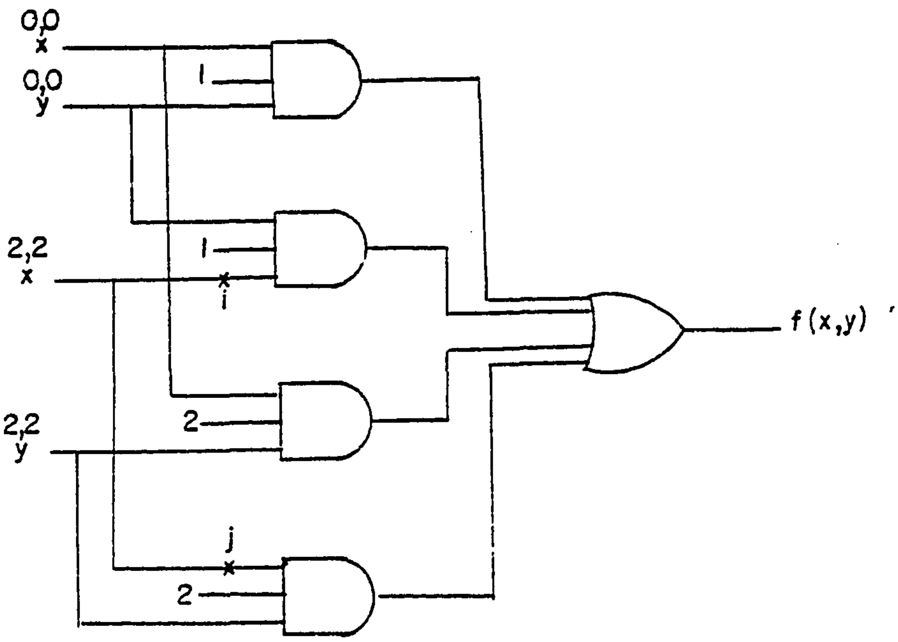


Figure 5.2.1: AND-OR Realization of the Function of Equation (5.2.7)

(a) fault on line i

Note that line i is connected to the input variable $x^{2,2}$ and the associated constant K is equal to 1. Therefore, the partial derivative must be taken with respect to $1\cdot x^{2,2}$.

$$\frac{\partial f}{\partial(1\cdot x^{2,2})} = (1\cdot x^{0,0}y^{0,0} + 1\cdot y^{0,0} + 2\cdot x^{0,0}y^{2,2} + 2\cdot x^{2,2}y^{2,2})$$
$$\oplus\ (1\cdot x^{0,0}y^{0,0} + 2\cdot x^{0,0}y^{2,2} + 2\cdot x^{2,2}y^{2,2}) = 1\cdot x^{1,2}y^{0,0} \qquad (5.2.8)$$

Set of tests T for detecting all s-a-0, s-a-1 and s-a-2 faults on line i are the values of x and y satisfying the following equation.

$$\frac{\partial f}{\partial(1\cdot x^{2,2})} = 1\cdot x^{1,2}y^{0,0} \leq 2$$

The above equation is satisfied when x = 1 and y = 0 or when x = 2 and y = 0, thus

T = [10,20]

Moreover, sets T_0 and T_k of tests for detecting s-a-0 and s-a-k faults on line i can be computed by using Equations (5.2.4) and (5.2.5), respectively, to Equation (5.2.8) as follows:

$$x^{2,2} \frac{\partial f}{\partial(1\cdot x^{2,2})} = (x^{2,2})(1\cdot x^{1,2}\, y^{0,0}) = 1\cdot x^{2,2}\, y^{0,0} \le 2 \tag{5.2.9}$$

$$\overline{x^{2,2}} \frac{\partial f}{\partial(1\cdot x^{2,2})} = (\overline{x^{2,2}})(1\cdot x^{1,2}\, y^{0,0}) = 1\cdot x^{1,1}\, y^{0,0} \le 2 \tag{5.2.10}$$

Equation (5.2.9) is satisfied when $x = 2$ and $y = 0$ and Equation (5.2.10) is satisified when $x = 1$ and $y = 0$, thus

$T_0 = [20]$ and $T_k = [10]$

Notice that $T = T_0 \cup T_k$ as expected, and s-a-1 and s-a-2 faults on line i are detectable, but they are indistinguishable because $K = 1$, this can be verified from the following table for set T_k.

T_k (x y)	f(x,y) (Normal)	f(x,y) (s-a-1)	f(x,y) (s-a-2)
1 0	0	1	1

(b) fault on line j

Line j is connected to the input variable $x^{2,2}$ and its associated constant $K = 2$. Thus, the partial derivative must be taken with respect to $2\cdot x^{2,2}$.

$$\frac{\partial f}{\partial(2\cdot x^{2,2})} = (1\cdot x^{0,0}\, y^{0,0} + 1\cdot x^{2,2}\, y^{0,0} + 2\cdot x^{0,0}\, y^{2,2} + 2\cdot y^{2,2})$$

$$\oplus\,(1\cdot x^{0,0}\,y^{0,0} + 1\cdot x^{2,2}\,y^{0,0} + 2\cdot x^{0,0}\,y^{2,2}) = 2\cdot x^{1,2}\,y^{2,2} \tag{5.2.11}$$

Set T is computed from Equation (5.2.11) as follows:

$$\frac{\partial f}{\partial (2\cdot x^{2,2})} = 2\cdot x^{1,2}\,y^{2,2} \leq 2$$

i.e., T = [12,22]

Set T_0 and T_k are computed from the following equations

$$x^{2,2}\,\frac{\partial f}{\partial (2\cdot x^{2,2})} = (x^{2,2})(2\cdot x^{1,2}\,y^{2,2}) = 2\cdot x^{2,2}\,y^{2,2} \leq 2$$

and

$$x^{\overline{2,2}}\,\frac{\partial f}{\partial (2\cdot x^{2,2})} = (x^{\overline{2,2}})(2\cdot x^{1,2}\,y^{2,2}) = 2\cdot x^{1,1}\,y^{2,2} \leq 2$$

i.e., T_0 = [22] and T_k = [12]

s-a-1 and s-a-2 faults on line j are detectable. They are also distinguishable because K = 2, this can be verified from the following table for set T_k.

T_k (x	y)	f(x,y) (Normal)	f(x,y) (s-a-1)	f(x,y) (s-a-2)
1	2	0	1	2

5.3 Single Faults Detection on Internal Lines

Single faults may occur on internal lines of multi-valued combinational circuits. When a single fault occurs on any internal line of a multi-valued combinational circuit, Theorem 5.2.2, Equation (5.2.4) and Equation (5.2.5) cannot be used for detecting this fault. For this case, the output function f of the given circuit must be written as function of the input variables $(x_1,\ldots,x_n)$ and function g_i, i.e.,

$$f = f(x_1,\ldots,x_n,g_i)$$

where g_i is function that must be read at line i and it is a function of the input variable $(x_1,\ldots,x_n)$ i.e., $g_i = g_i(x_1,\ldots,x_n)$

The following three lemmas will be used for detecting single fault on any internal line of a multi-valued combinational circuit:

Lemma 5.3.1:

Set of tests T for detecting both s-a-0 and s-a-k faults on an internal line i is computed from the following equation.

$$\left.\frac{\partial f}{\partial(K\cdot g_i)}\right|_{a_1\ldots a_n} \leq L \qquad 0 < L \leq m-1 \qquad (5.3.1)$$

<u>Lemma 5.3.2</u>:

Set of tests T_0 for detecting s-a-0 fault on an internal line i is computed from the following equation:

$$g_i \left. \frac{\partial f}{\partial (K\cdot g_i)} \right|_{a_1 \ldots a_n} \leq L \qquad 0 < L \leq m-1 \qquad (5.3.2)$$

<u>Lemma 5.3.3</u>:

Set of tests T_k for detecting s-a-k fault on an internal line i is computed from the following equation:

$$\overline{g}_i \left. \frac{\partial f}{\partial (K\cdot g_i)} \right|_{a_1 \ldots a_n} \leq L \qquad 0 < L \leq m-1 \qquad (5.3.3)$$

<u>Example 5.3.1</u>

Consider the 5-valued combinational circuit shown in Figure 5.3.1 whose output function $f(x_1,x_2,x_3)$ is given by Equation (5.3.4). It is desired to compute sets T, T_0 and T_k for detecting all faults on line i indicated in the figure.

$$f(x_1,x_2,x_3) = (1\cdot x_1^{4,4}\, x_3^{0,0} + 3\cdot x_2^{4,4}\, x_3^{\overline{0,0}})(4\cdot x_2^{4,4} + 2\cdot x_1^{2,3}\, x_3^{1,3}) \qquad (5.3.4)$$

function f can be written as a function of x_1,x_2,x_3 and g_i as follows:

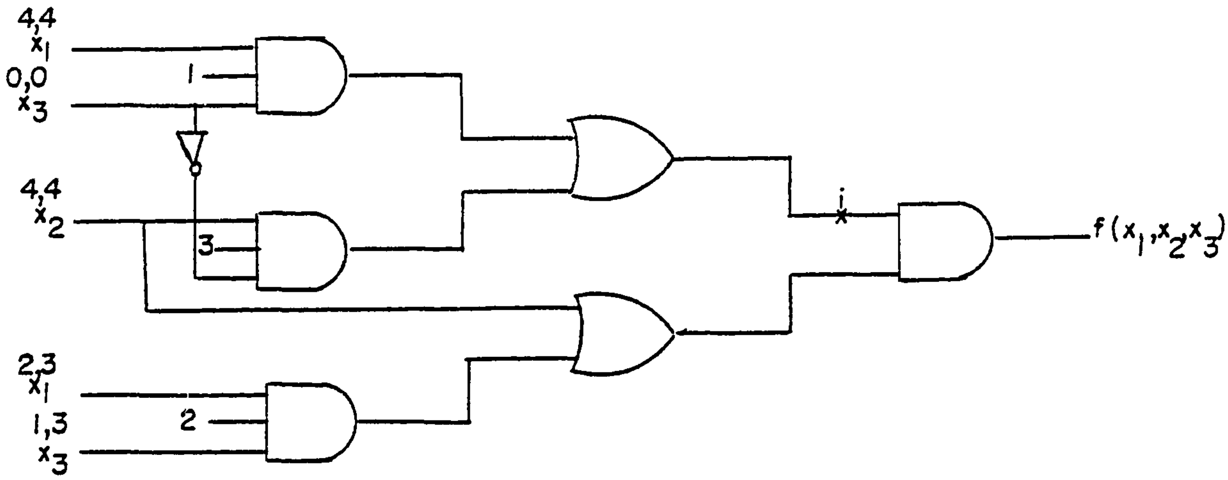


Figure 5.3.1: Circuit of Example 5.3.1

$$f(x_1,x_2,x_3,g_i) = (4\cdot g_i)(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3})$$

$$\text{where } g_i = 1\cdot \overset{4,4}{x_1}\ \overset{0,0}{x_3} + 3\cdot \overset{4,4}{x_2}\ \overset{\overline{0,0}}{x_3}$$

$$\frac{\partial f}{\partial(4\cdot g_i)} = g_i(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3}) \oplus \overline{g}_i(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3}) \tag{5.3.5}$$

$$\begin{aligned}\frac{\partial f}{\partial(4\cdot g_i)} &= (1\cdot \overset{4,4}{x_1}\ \overset{0,0}{x_3} + 3\cdot \overset{4,4}{x_2}\ \overset{\overline{0,0}}{x_3})(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3}) \\ &\quad \oplus \overline{(1\cdot \overset{4,4}{x_1}\ \overset{0,0}{x_3} + 3\cdot \overset{4,4}{x_2}\ \overset{\overline{0,0}}{x_3})}(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3}) \\ &= (1\cdot \overset{4,4}{x_1}\ \overset{0,0}{x_3} + 3\cdot \overset{4,4}{x_2}\ \overset{\overline{0,0}}{x_3})(4\cdot \overset{4,4}{x_2} + 2\cdot \overset{2,3}{x_1}\ \overset{1,3}{x_3})\end{aligned}$$

$$\oplus\ \overline{(1\cdot x_1^{4,4}\ x_3^{\overline{0,0}})}\ \overline{(3\cdot x_2^{4,4}\ x_3^{0,0})}\ (4\cdot x_2^{4,4} + 2\cdot x_1^{2,3}\ x_3^{1,3})$$

$$= (1\cdot x_1^{4,4}\ x_3^{0,0} + 3\cdot x_2^{4,4}\ x_3^{\overline{0,0}})(4\cdot x_2^{4,4} + 2\cdot x_1^{2,3}\ x_3^{1,3})$$

$$\oplus\ (3 + 4\cdot x_1^{\overline{4,4}} + 4\cdot x_3^{\overline{0,0}})(1 + 4\cdot x_2^{\overline{4,4}} + 4\cdot x_3^{0,0})(4\cdot x_2^{4,4} + 2\cdot x_1^{2,3}\ x_3^{1,3})$$

$$= 2\cdot x_1^{2,3}\ x_2^{\overline{4,4}}\ x_3^{1,3} + 3\cdot x_1^{4,4}\ x_2^{4,4}\ x_3^{0,0} + 4\cdot x_1^{\overline{4,4}}\ x_2^{4,4}$$

$$+ 4\cdot x_1^{4,4}\ x_2^{4,4}\ x_3^{\overline{0,0}} \tag{5.3.6}$$

Set T can be computed by making equation (5.3.6) equal to or less than 4, thus

$$\frac{\partial f}{\partial(4\cdot g_i)} = 2\cdot x_1^{2,2}\ x_2^{\overline{4,4}}\ x_3^{1,3} + 3\cdot x_1^{4,4}\ x_2^{4,4}\ x_3^{0,0} + 4\cdot x_1^{\overline{4,4}}\ x_2^{4,4}$$

$$+ 4\cdot x_1^{4,4}\ x_2^{4,4}\ x_3^{\overline{0,0}} \leq 4 \tag{5.3.7}$$

By solving Equation (5.3.6), set T can be obtained as:

T = [040,041,042,043,044,140,141,142,143,144,201, 202,203,211,212,213,221,222,223,231,232,233,240, 241,242,243,244,301,302,303,311,312,313,321,322, 323,331,332,333,340,341,342,343,344,440,441,442, 443,444]

Sets T_0 and T_k can be computed by using Equations (5.3.2) and (5.3.3) to Equation (5.3.6), respectively, as follows:

$$g_i \frac{\partial f}{\partial(4\cdot g_i)} = (1\cdot x_1^{4,4} x_3^{0,0} + 3\cdot x_2^{4,4} x_3^{\overline{0,0}})(2\cdot x_1^{2,3} x_2^{\overline{4,4}} x_3^{1,3}$$
$$+ 3\cdot x_1^{4,4} x_2^{4,4} x_3^{0,0} + 4\cdot x_1^{\overline{4,4}} x_2^{4,4} + 4\cdot x_1^{4,4} x_2^{4,4} x_3^{\overline{0,0}}) \leq 4 \tag{5.3.8}$$

$$\overline{g_i} \frac{\partial f}{\partial(4\cdot g_i)} = (1 + 3\cdot x_2^{4,4} + 3\cdot x_3^{0,0} + 1\cdot x_1^{\overline{4,4}} + 4\cdot x_1^{\overline{4,4}} x_2^{4,4} + 4\cdot x_1^{4,4} x_3^{0,0}$$
$$+ 1\cdot x_3^{\overline{0,0}} + 4\cdot x_2^{4,4} x_3^{\overline{0,0}})(2\cdot x_1^{2,3} x_2^{\overline{4,4}} x_3^{1,3} + 3\cdot x_1^{4,4} x_2^{4,4} x_3^{0,0}$$
$$+ 4\cdot x_1^{\overline{4,4}} x_2^{4,4} + 4\cdot x_1^{4,4} x_2^{4,4} x_3^{\overline{0,0}}) \leq 4 \tag{5.3.9}$$

By solving Equations (5.3.8) and (5.3.9), sets T_0 and T_k can be obtained:

T_0 = [041,042,043,141,142,143,144,241,242,243,244, 341,342,343,344,440,441,442,443,444]

T_k = [040,041,042,043,044,140,141,142,143,144,201, 202,203,211,212,213,221,222,223,231,232,233,240, 241,242,243,244,301,302,303,311,312,313,321,322, 323,331,332,333,340,341,342,343,344,440,441,442, 443,444]

CHAPTER VI

STATIC HAZARD DETECTION IN ASYNCHRONOUS MULTI-LEVEL COMBINATIONAL CIRCUITS

6.1 Introduction

Since the first investigation by Huffman [32] and Eichelberger [33], the problem of static hazard detection in two-valued asynchronous combinational circuits has been attacked by many researchers [34-39]. Eichelberger [33], followed by Bredson and Hulina [34], Mukcaidono [35], Mete et al. [36], and Thayse [37] studied the problem by assuming that the changes of two or more input signals must happen simultaneously. By carefully reviewing practical situations, it is found that to control the timing of the changes of two or more input signals with the presence of random delays in an asynchronous circuit is extremely difficult, if not totally impossible. Therefore, in this dissertation, we eliminate this unrealistic assumption and consider static hazards under single input change at a time.

It is also important to point out that most of the previous works on this subject [33-38] were generally limited to two-level combinational circuit hazard detection. Since static 0 (static 1) hazard cannot occur in two-level AND-OR

(OR-AND) asynchronous circuits, the problem of static hazard detection of two-level circuits is considerably simpler than that of multi-level cicuits where both static 0 and static 1 hazards may occur.

In this chapter we present a procedure for static (logic) hazard detection in asynchronous multi-level multi-valued combinational circuits using the multi-valued digital calculus introduced in Chapter IV. The procedure uses the transforming of the given multi-level circuit into its equivalent two-level circuit, called equivalent normal form (ENF) [40] and is derived from the concept of multi-valued partial derivative.

<u>Definition 6.1.1</u>

Define the initial input state $A = (b_1,\ldots,b_{j-1},c_j, b_{j+1},\ldots,b_n)$ and the final input state $B = (b_1,\ldots,b_{j-1},b_j, b_{j+1},\ldots,b_n)$, where $c_j \neq b_j$.

6.2 Types of Static Hazard

Just like static hazards exist in binary circuits; so does it exist in multi-valued circuits. There exists two types of static hazards in multi-valued circuits, the first type is called input static hazard (function static hazard) and the second type called logic static hazard. An input static hazard is caused by the presence of a spurious pulse in the output signal, due to a transient input state occuring during the transition from an input state A to an input state B.

The input variable x_i attains this wrong transient value momentary. The spurious hazard pulse appears at the output if the difference between the given values of x_i in state A and state B is greater than one. If $x_i = p$ in state A and $x_i = p \pm 1$ in state B, the transition of x_i from state A to state B will not contain a middle value and thus no transient state may occur during this transition. Hence a practical multi-valued combinational circuit is free of all input static hazards if the input variables $x_i = p$ before the change and $x_i = p \pm 1$ after the change. The circuits under this mode of operation are reliable and thus useful for practical applications. Therefore, in this dissertation, we consider only this mode of operation. The second type of hazards is logic static hazard and it is defined in the following definition.

<u>Definition 6.2.1</u>

A multi-valued combinational circuit contains logic static hazard or simply static hazard if and only if

(1) $f(A) = f(B)$, where A and B were defined in Definition 6.1.1.

(2) during the input change A to B a spurious hazard pulse may be present on the output.

Static hazard (logic static hazard) is present in a multi-valued combinational circuit if one or more of the input variables and their complements are both present in the realization of the circuit. Any input variable and its complement may not change exactly at the same time and thus the circuit may momentarily give a spurious pulse in the output signal, this spurious pulse at the output causes static hazard to be present.

6.3 Combintorial Variables

The static hazard is closely related to the internal propogation times of the signal path, i.e., on the internal delays of the invertors in the circuit. The presence of these delays in a circuit requests to treat internal switching signals as independent variables during an input transition. So to make the internal signals as independent, the circuit must not have any two parallel paths. The parallel paths are those paths that are branching from the same node and meet at another single node. The introduction of combintorial variables in the circuit eliminates the existance of parallel paths and make the switching signals independent during an input transition. However, the introduction of the combintorial variables will not affect the truth table of the circuit.

A large number of combintorial variables being added to the circuit makes the analysis procedure very tedious, therefore, it is desirable to determine the minimal set of combintorial variables that can do the job. The determination of the minimal set of combintorial variables will be illustrated by examples in section 6.5.

It is obvious that the set of input variables (${}^{a_1,b_1}x_1$, ..., ${}^{a_n,b_n}x_n$) is transformed into the set of input and combintorial variable (${}^{a_1,b_1}x_1,\ldots,{}^{a_n,b_n}x_n$, $w_1,\ldots,w_\ell$) and the output function $z = f({}^{a_1,b_1}x_1,\ldots,{}^{a_n,b_n}x_n)$ is transformed into $z = f({}^{a_1,b_1}x_1,\ldots,{}^{a_n,b_n}x_n,w_1,\ldots,w_\ell)$. The number of the outputs of the modified circuit became $1 + \ell$ where ℓ is the number of combintorial variables that are being added. ${}^{a_i,b_i}x_i$ can be more than one variable depends on whether a_i and b_i are vectors or have one value.

6.4 Static Hazard Detection Using Digital Calculus

In order to detect the static hazard in multi-valued combinational circuits using multi-valued digital calculus that was introduced in Chapter IV, the differentials df and dw_j for $j = 1,\ldots,\ell$ must be calculated to predict the transient behavior of the circuit as it was suggested by Thayse [37] for two-valued case only. The combintorial variables w_1 is called the combintorial variable of order 1 because they are only considered as the functions of the input variables, w_2 is called the combintorial variables of order 2 because they are considered as the functions of the input variables and the combintorial variables of order 1, and so on. In general w_j is called the combintorial variables of order j.

$$w_j = w_j({}^{a_1,b_1}x_1,\ldots,{}^{a_n,b_n}x_n,w_1,\ldots,w_{j-1}) \qquad (6.4.1)$$

The differentials dw_j for $j = 1,\ldots,\ell$, and df are given by

$$dw_j = \sum_{i=1}^{n} \frac{\partial w_j}{\partial^{a_i,b_i} x_i} \, d^{a_i,b_i} x_i + \sum_{k=1}^{\ell} \frac{\partial w_j}{\partial w_k} \, dw_k \quad k \neq j \tag{6.4.2}$$

$$df = \sum_{i=1}^{n} \frac{\partial f}{\partial^{a_i,b_i} x_i} \, d^{a_i,b_i} x_i + \sum_{k=1}^{\ell} \frac{\partial f}{\partial w_k} \, dw_k \tag{6.4.3}$$

The way to evaluate Equations (6.4.2) and (6.4.3) is to first compute the differentials of the combintorial variables of order 1. If some of these differentials are not equal to zero, they are made equal to m-1 in order to compute the differentials of order 2. This process is then continued for at most ℓ steps. One then has to compute those w_j's that their differentials are equal to zero. Finally df can be computed.

The following theorem will be useful for static hazard detection of multi-valued combinational circuits.

<u>Theorem 6.4.1</u>

If f(A) = f(B), the realization of the function f(x,w) is free of static hazard if and only if:

$$df(x,w) = \left[\sum_{i=1}^{n} \frac{\partial f}{\partial^{a_i,b_i} x_i} \, d^{a_i,b_i} x_i \right]_C + \left[\sum_{k=1}^{\ell} \frac{\partial f}{\partial w_k} \, dw_k \right]_C = 0 \tag{6.4.4}$$

where $d\,{}^{a_i,b_i}x_i\,(dw_k) = m - 1$ if ${}^{a_i,b_i}x_i\,(w_k)$ changes

$= 0$ if ${}^{a_i,b_i}x_i\,(w_k)$ does not change

and $C = (b_1,\ldots,b_{j-1},b_{j+1},\ldots b_n)$

Proof: It will be shown that if either summations of Equation (6.4.4) of Theorem 6.4.1 is not satisfied, the function f will contain a static hazard for considered transition. If Equation (6.4.4) is not satisfied, then there exists at least one term in either summations is not equal to zero, e.g., the jth term of the first summation is not equal to zero which is given by Equation (6.4.5).

$$df_j = [f(x_1,\ldots,x_n,\ {}^{a_j,b_j}x_j\ )\ \oplus\ f(x_1,\ldots,x_n,\ {}^{\overline{a_j,b_j}}x_j\ )]\ d\ {}^{a_j,b_j}x_j \tag{6.4.5}$$

But from the definition of the multi-valued partial derivative if both terms of Equation (6.4.5) are not equal, then the function f experiences a variation for the jth term of the first summation for the considered transition. Also $d\,{}^{a_j,b_j}x_j$ is not equal to zero because the values of ${}^{a_j,b_j}x_j$ in state A and state B are not equal. This concludes that f contains static hazard.

6.5 Procedure for Detecting Static Hazards

A procedure for detecting static hazard in a multi-valued combinational circuit by first obtaining the equivalent normal form (ENF) [40] of the given circuit is given below:

(1) Find the equivalent normal form (ENF) of the given multi-level multi-valued combinational circuit.

(2) Find all parallel paths from the ENF

(3) Find all minimal sets of combintorial variables by using the minimal covering technique such as Quine-McClusky covering table.

(4) Choose arbritrary one set that obtained in (3).

(5) Construct a free parallel paths circuit by including the combintorial variables obtained in (4).

(6) Write $1 + \ell$ equations of the output functions of the circuit obtained in (5).

(7) Compute the differentials dw_j for $j = 1,\ldots,\ell$, and df.

(8) Use Theorem 6.4.1 to find the transitions that contain static hazard.

It is found that the equivalent normal form (ENF) can be used to determine the parallel paths of the given circuit. Equivalent normal form (ENF) is illustrated in the following example.

Example 6.5.1

Consider the following 3-level, 3-valued switching circuit shown in Figure 6.5.1. Find the minimal sets of comintorial variables of this circuit.

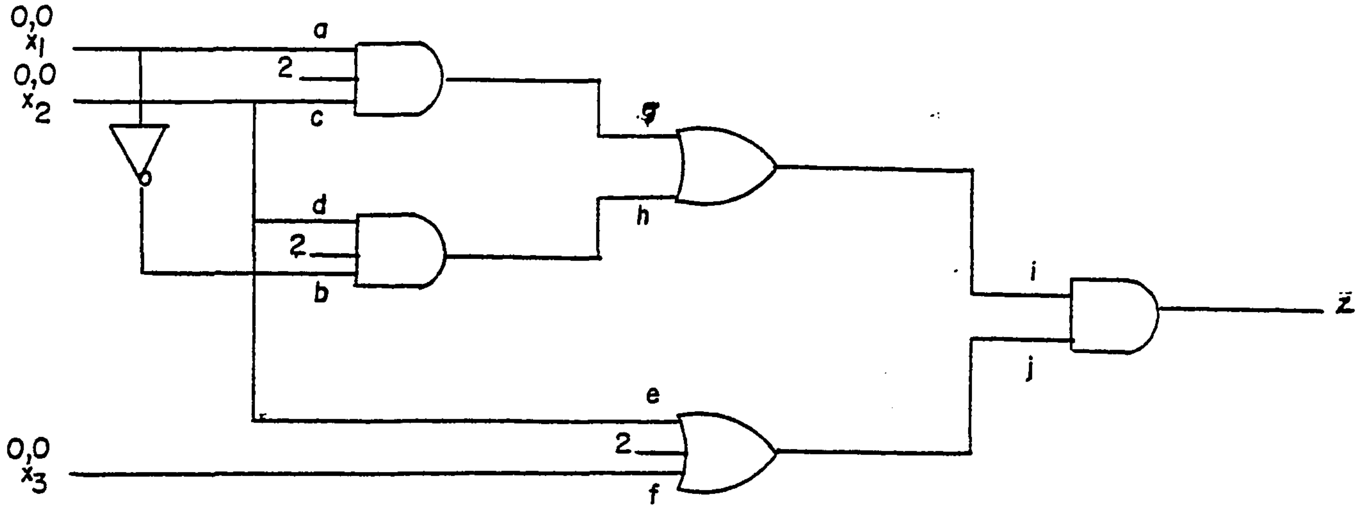


Figure 6.5.1: Circuit of Example 6.5.1.

The ENF of the circuit of Figure 6.5.1 can be obtained as follows:

$$z = i \cdot j = (g + h)_i \ (e + f)_j = (g_i + h_i)(e_j + f_j)$$

$$= g_i \ e_j + g_i \ f_j + h_i \ e_j + h_i \ f_j$$

$$= (2ac)_{gi} \ e_j + (2ac)_{gi} \ f_j + (2bd)_{hi} \ e_j + (2bd)_{hi} \ f_j$$

$$= 2a_{gi} \ c_{gi} \ e_j + 2a_{gi} \ c_{gi} \ f_j + 2b_{hi} d_{hi} \ e_j + 2b_{hi} d_{hi} \ f_j$$

$$= 2 \ x_{1_{agi}}^{0,0} \ x_{2_{cgi}}^{0,0} \ x_{2_{ej}}^{0,0} + 2 \ x_{1_{agi}}^{0,0} \ x_{2_{cgi}}^{0,0} \ x_{3_{fj}}^{0,0}$$

$$+ 2 \; \overline{\overset{0,0}{x}}_{1_{bhi}} \; \overset{0,0}{x}_{2_{dhi}} \; \overset{0,0}{x}_{2_{ej}} + 2 \; \overline{\overset{0,0}{x}}_{1_{dhi}} \; \overset{0,0}{x}_{2_{dhi}} \; \overset{0,0}{x}_{3_{fj}} \tag{6.5.1}$$

All possible parallel paths of the circuit of Figure 6.5.1 can be found from Equation (6.5.1) by using the following two steps:

Step 1: Search for each variable and its complement, if present, in each term of Equation (6.5.1) and list all the paths associated with this variable and its complement to form one set of parallel paths. For example, the variable $\overset{0,0}{x}_1$ and its complement $\overline{\overset{0,0}{x}}_1$ give the set of parallel paths [(a,g,i),(b,h,i)]. Also, the variable $\overset{0,0}{x}_2$ gives the sets of parallel paths [(c,g,i),(e,j),(d,h,i)]. Notice that both (c,g,i) and (e,j) are both present in one term of Equation (6.5.1).

Step 2: The two sets [(a,g,i),(b,h,i)] and [(c,g,i),(e,j),(d,h,i)] contain only the longest paths, i.e., the paths start at the input of the circuit and end at its output. But we also need to find all sub-parallel paths present in the circuit. It is found that the searching tree method is very convenient to find these sub-parallel paths. One searching tree is needed for each set of parallel path obtained in step 1. The searching tree method can be summarized as follows:

(1) Insert one set of the longest parallel paths that are obtained in step 1 in the first level of the tree.

(2) All subsets that have at least two parallel paths of the set of the first level of the tree are inserted in the second level of the tree. The number of these subsets can be computed by using the following formula:

$$\binom{r}{i} = \frac{r(r-1)(r-2)\ldots(r-i)}{i!}$$

where r = the number of longest parallel paths that were obtained in step 1

$i = 2,3,\ldots,r-1$

Let q be the total number of subsets of sub-parallel paths in the second level of the tree or be the total number of braches of the second level of the tree.

$$q = \binom{r}{2} + \binom{r}{3} + \ldots + \binom{r}{r-1}$$

$$q = \sum_{i=2}^{r-1} \binom{r}{i} \qquad (6.5.2)$$

where $\binom{r}{i}$ is the number of branches that contain i sub-parallel paths.

If the set of the first level of the tree contains only two parallel paths, then go to (3).

(3) If all parallel paths in any subset of part (2) have one common liter at the end, delete this common liter in all paths of the subset and insert this new subset of sub-parallel paths in the next level of the tree.

(4) Any branch of the tree is terminated when there is no common literals in all paths of the branch.

Searching trees for sets [(a,g,i),(b,h,i)] and [(e,g,i),(e,j),(d,h,i)] are shown in Figure 6.5.2a and Figure 6.5.2b, respectively. Notice that r=3, i=2 and q=3 for the set [(e,g,i) (e,j), (d,h,i)] of the searching tree of Figure 6.5.2b.

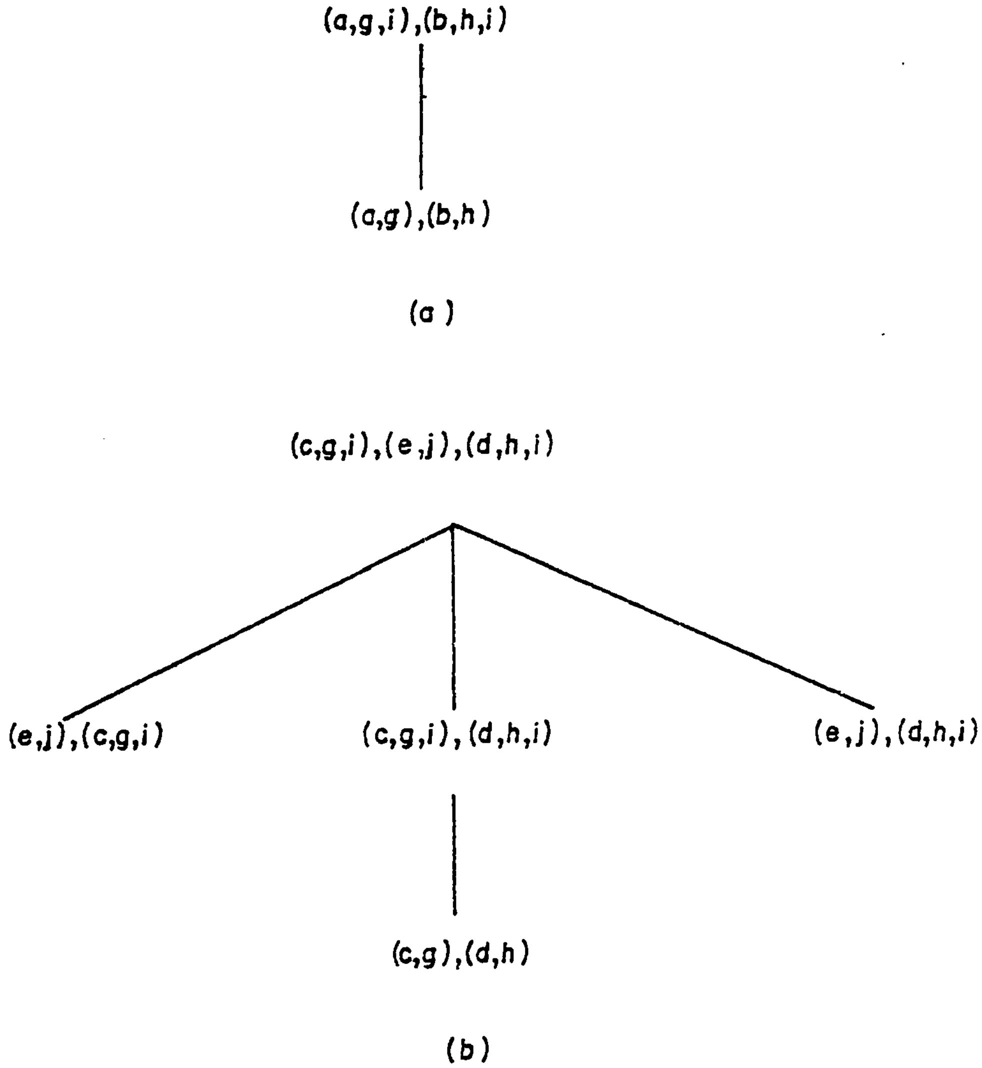


Figure 6.5.2: Searching Trees for Example 6.5.1.

The complete sets of parallel paths of the circuit of Figure 6.5.1 can now be obtained from the searching trees of Figure 6.5.2, these sets are:

(a,g,i) and (b,h,i)

(a,g) and (b,h)

(c,g,i),(e,j) and (d,h,i)

(e,j) and (c,g,i)

(c,g,i) and (d,h,i)

(e,j) and (d,h,i)

(c,g) and (d,h)

The problem of determining the minimus sets of combintorial variables becomes a minimal covering problem as shown in Table 6.5.1. Each column of Table 6.5.1 corresponds to one of the seven sets of parallel paths obtained above and each row corresponds to one letter that is shown at the input of the gates of Figure 6.5.1.

TABLE 6.5.1

MINIMAL COVERING TABLE FOR EXAMPLE 6.5.1

	(abghi)	(abgh)	(cdeghij)	(cdgh)	(cegij)	(dehij)	(cdghi)
a	x	x					
b	x	x					
c			x	x	x		x
d			x	x		x	x
e			x		x	x	
f							
g	x	x	x	x	x		x
h	x	x	x	x		x	x
i	x		x		x	x	x
j			x		x	x	

From Table 6.5.1, the minimal sets are:

(c,h),(d,g),(e,g),(e,h),(g,h),(g,i),(g,j),(h,i), and (h,j)

Choose arbritrarily the minimal set (g,h). The circuit that is free of all parallel paths in which it includes the combintorial variables w_g and w_h is shown in Figure 6.5.3.

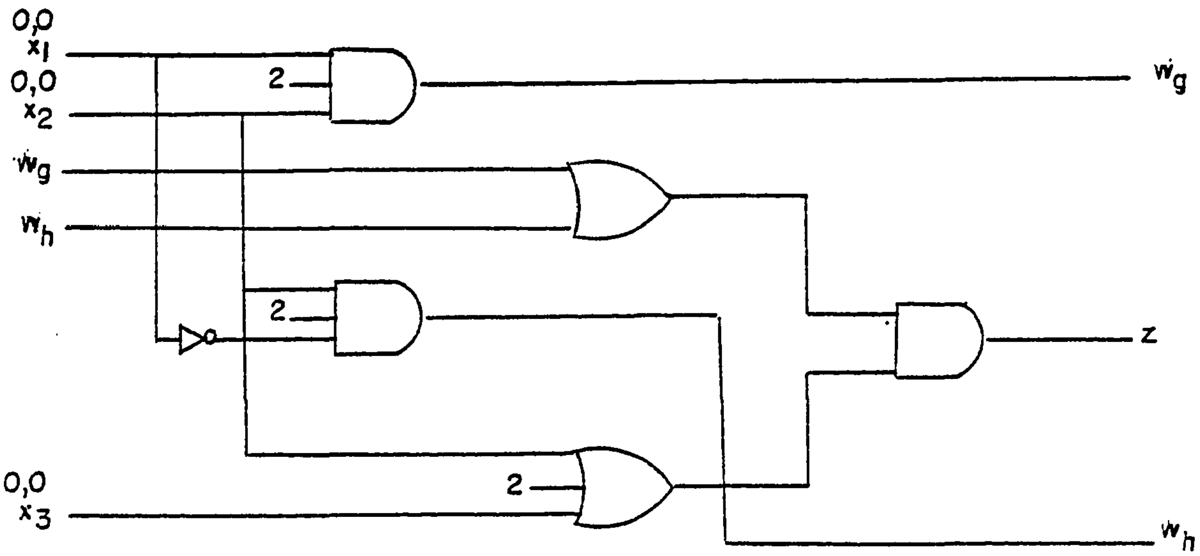


Figure 6.5.3: The circuit of Figure 6.5.1 without parallel paths.

The outputs of the circuit of Figure 6.5.3 are:

$$w_g = 2.\ \overset{0,0}{x_1}\ \overset{0,0}{x_2} \tag{6.5.3}$$

$$w_h = 2.\ \overset{\overline{0,0}}{x_1}\ \overset{0,0}{x_2} \tag{6.5.4}$$

$$z = (w_g + w_h)(2.\ \overset{0,0}{x_2} + 2.\ \overset{0,0}{x_3}) \tag{6.5.5}$$

Example 6.5.2

Detect static hazards present in the circuit of Figure 6.5.1.

Free parallel paths circuit of the circuit of Figure 6.5.1 is shown in Figure 6.5.3. The differentials dw_g, dw_h and dz of Equations (6.5.3), (6.5.4) and (6.5.5) are computed as follows:

$$dw_g = \frac{\partial w_g}{\partial \overset{0,0}{x_1}}\ d\overset{0,0}{x_1} + \frac{\partial w_g}{\partial \overset{0,0}{x_2}}\ d\overset{0,0}{x_2}$$

$$= (2\ \overset{0,0}{x_2} \oplus 0)\ d\overset{0,0}{x_1} + (2\ \overset{0,0}{x_1} \oplus 0)\ d\overset{0,0}{x_2}$$

$$= 2.\ \overset{0,0}{x_2}\ d\overset{0,0}{x_1} + 2.\ \overset{0,0}{x_1}\ d\overset{0,0}{x_2}$$

$$dw_h = \frac{\partial w_h}{\partial \overset{0,0}{x_1}}\ d\overset{0,0}{x_1} + \frac{\partial w_h}{\partial \overset{0,0}{x_2}}\ d\overset{0,0}{x_2}$$

$$= (0 \oplus 2.\ \overset{0,0}{x_2})d\overset{0,0}{x_1} + (2.\ \overset{\overline{0,0}}{x_1} \oplus 0)d\overset{0,0}{x_2}$$

$$= 2.\ \overset{0,0}{x_2}\ d\overset{0,0}{x_1} + 2.\ \overset{\overline{0,0}}{x_1}\ d\overset{0,0}{x_2}$$

$$dz = \frac{\partial z}{\partial \overset{0,0}{x_2}}\ d\overset{0,0}{x_2} + \frac{\partial z}{\partial \overset{0,0}{x_3}}\ d\overset{0,0}{x_3} + \frac{\partial z}{\partial w_g}\ dw_g + \frac{\partial z}{\partial w_h}\ dw_h$$

$$= [(w_g + w_h)(2 + 2.\ \overset{0,0}{x_3}) \oplus (w_g + \overline{w}_h)(0 + 2\ \overset{0,0}{x_3})]d\overset{0,0}{x_2}$$

$$+\ [(w_g + w_h)(2.\,{}^{0,0}x_2 + 2) \oplus (w_g + w_h)(2.\,{}^{0,0}x_2 + 0)]\,d\,{}^{0,0}x_3$$

$$+\ [(w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3) \oplus (\overline{w}_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3)]\,dw_g$$

$$+\ [(w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3) \oplus (w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3)]\,dw_h$$

$$dz = [(w_g + w_h) \oplus 2.\,{}^{0,0}x_3\,(w_g + w_h)]\,d\,{}^{0,0}x_2$$

$$+\ [(w_g + w_h) \oplus 2.\,{}^{0,0}x_2\,(w_g + w_h)]\,d\,{}^{0,0}x_3$$

$$+\ [(w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3) \oplus (\overline{w}_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3)]\,dw_g$$

$$+\ [(w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3) \oplus (w_g + w_h)(2.\,{}^{0,0}x_2 + 2.\,{}^{0,0}x_3)]\,dw_h$$

To test the transition from state (001) to state (002) for static hazard, it can easily be verified that $d\,{}^{0,0}x_1 = 0$, $d\,{}^{0,0}x_2 = 0$, $dw_g = 0$, $dw_h = 0$, $w_g = 2$, and $w_h = 0$. By substituting these values into dz, we obtain:

$$dz = [(2+0) \oplus 2\cdot{}^{0,0}x_3\,(2+0)](0) + [(2+0) \oplus 2\cdot(2+0)]\,d\,{}^{0,0}x_3$$

$$+\ [(2+0)(2 + 2\cdot{}^{0,0}x_3) \oplus (0 + 0)(2 + 2\cdot{}^{0,0}x_3)](0)$$

$$+\ [(2+0)(2 + 2\cdot{}^{0,0}x_3) \oplus (2 + 2)(2 + 2\cdot{}^{0,0}x_3)](0)$$

$$= (2 \oplus 2)\,d\,{}^{0,0}x_3 = (0)\,d\,{}^{0,0}x_3 = 0$$

Thus $dz = 0$, which implies that this transition is free of static hazard according to Theorem 6.4.1. On the other hand, the transition from state (002) to state (102) yields $d^{0,0}x_2 = 0$, $d^{0,0}x_3 = 0$, $dw_g = 2$, and $dw_h = 2$. By substituting these values into dz, we obtain that $dz = 2$, i.e., this transition is not free of static hazard. Table 6.5.2 shows all static hazards that are present in the circuit of Figure 6.5.1.

TABLE 6.5.2

Static Hazard Table for Example 6.5.2

A	B	f(A)=f(B)	dw_g	dw_h	w_g	w_h	dz	Type of Hazard
000	100	2	2	2			2	2-hazard
001	101	2	2	2			2	2-hazard
002	102	2	2	2			2	2-hazard

<u>Example 6.5.3</u>

Consider the following 4-level, 5-valued switching circuit shown in Figure 6.5.4. Detect all static hazards present in this circuit.

Applying static hazard detection procedures, we obtain ENF of the circuit which is given by Equation (A.1.2) in Section A.1 of Appendix A.

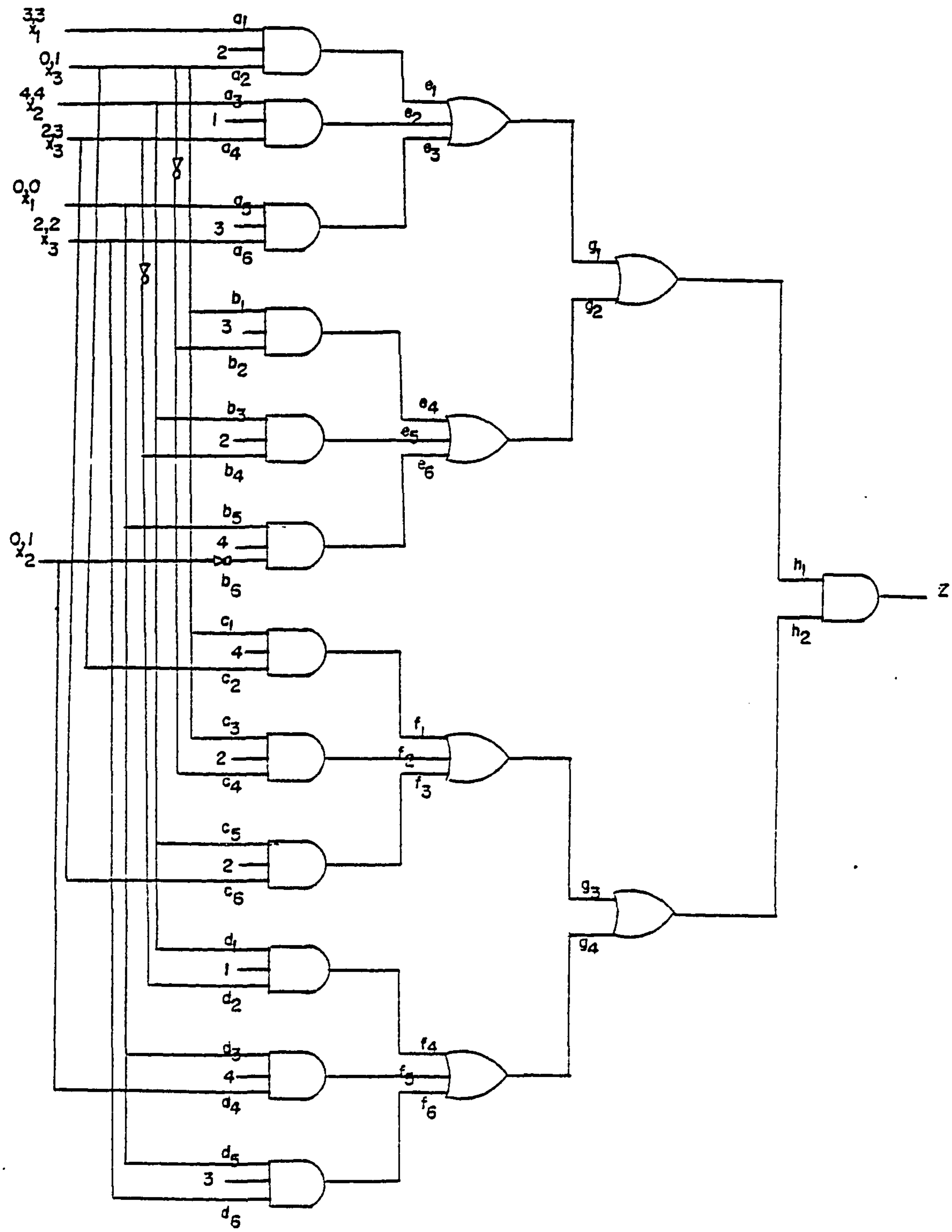


Figure 6.5.4: Circuit of Example 6.5.3

The minimal sets of parallel paths are given in section A.2 of Appendix A. Choose arbritarily from the minimal sets that are obtained in section A.2 the minimal set $(f_1, f_6, g_2, g_3, h_1)$. The circuit that is free of all parallel paths in which it includes the combintorial variables w_{f_1}, w_{f_6}, w_{g_2}, w_{g_3}, w_{h_1} is shown in Figure 6.5.5.

The outputs of the circuit for Figure 6.5.5 are:

$$w_{f_1} = 4 \cdot x_1^{3,3} x_3^{0,1}$$

$$w_{f_6} = 3 \cdot x_1^{0,0} x_3^{2,2}$$

$$w_{g_2} = 3 \cdot x_1^{3,3} x_3^{\overline{0,1}} + 2 \cdot x_2^{4,4} x_3^{\overline{2,3}} + 4 \cdot x_1^{0,0} x_2^{\overline{0,1}}$$

$$w_{g_3} = w_{f_1} + 2 \cdot x_1^{3,3} x_3^{\overline{0,1}} + 2 \cdot x_2^{4,4} x_3^{2,3}$$

$$w_{h_1} = w_{g_2} + 2 \cdot x_1^{3,3} x_3^{0,1} + 1 \cdot x_2^{4,4} x_3^{2,3} + 3 \cdot x_1^{0,0} x_3^{2,2}$$

$$z = w_{h_1} (w_{g_3} + 1 \cdot x_2^{4,4} x_3^{\overline{2,3}} + 4 \cdot x_1^{0,0} x_2^{0,1} + w_{f_6})$$

$$= w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 1 \cdot x_2^{4,4} x_3^{\overline{2,3}} w_{h_1} + 4 \cdot x_1^{0,0} x_2^{0,1} w_{h_1}$$

The differentials dw_{f_1}, dw_{f_6}, dw_{g_2}, dw_{g_3} and dw_{h_1}, and dz are computed as follows:

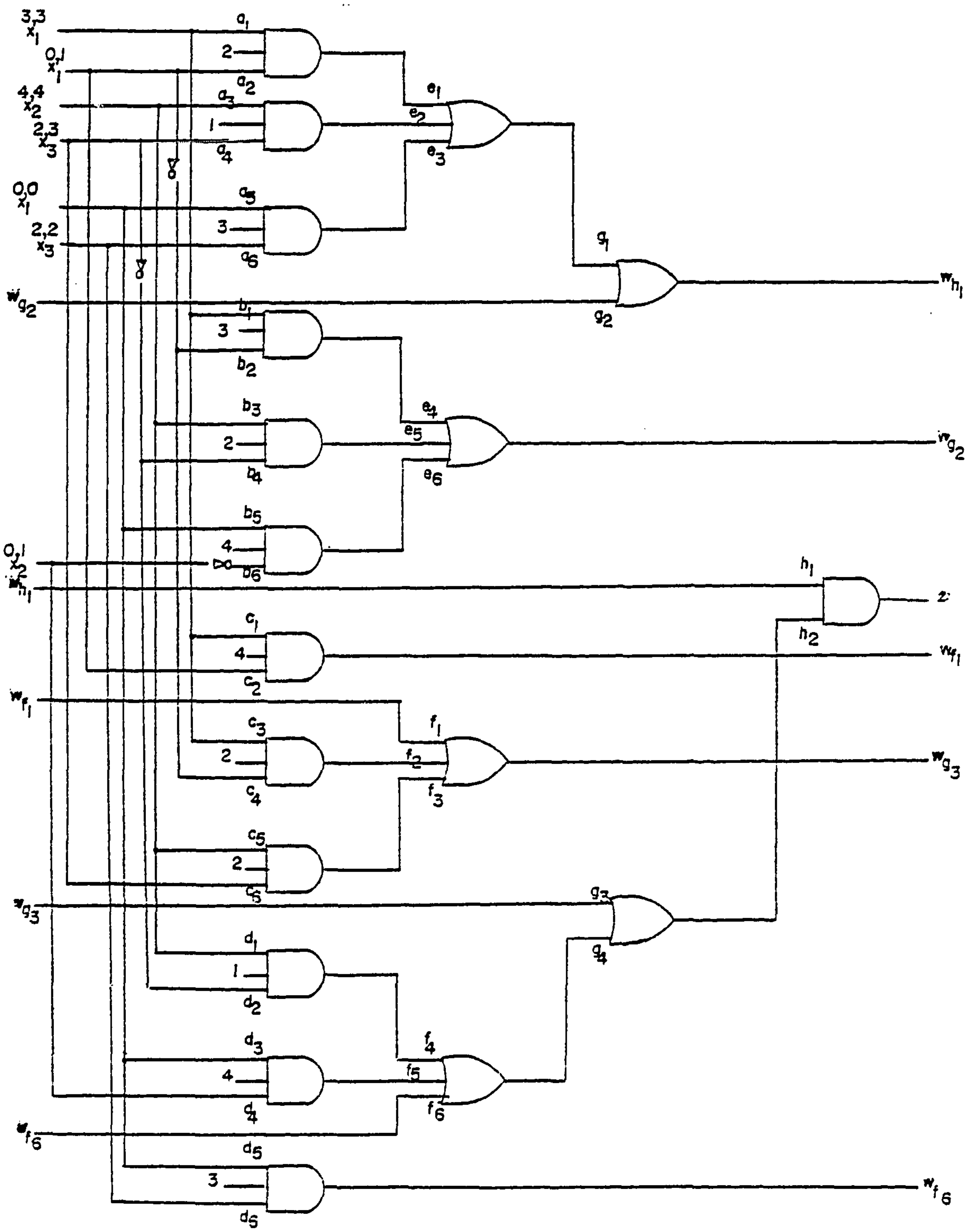


Figure 6.5.5: The Circuit of Figure 6.5.4 Without Parallel Paths

$$dw_{f_1} = \frac{\partial w_{f_1}}{\partial x_1^{3,3}} d\, x_1^{3,3} + \frac{\partial w_{f_1}}{\partial x_3^{0,1}} d\, x_3^{0,1}$$

$$= 4\cdot x_3^{0,1}\, d\, x_1^{3,3} + 4\cdot x_1^{3,3}\, d\, x_3^{0,1}$$

$$dw_{f_6} = \frac{\partial w_{f_6}}{\partial x_1^{0,0}} d\, x_1^{0,0} + \frac{\partial w_{f_6}}{\partial x_3^{2,2}} d\, x_3^{2,2}$$

$$= 3\cdot x_3^{2,2}\, d\, x_1^{0,0} + 3\cdot x_1^{0,0}\, d\, x_3^{2,2}$$

$$dw_{g_2} = \frac{\partial w_{g_2}}{\partial x_1^{3,3}} d\, x_1^{3,3} + \frac{\partial w_{g_2}}{\partial x_1^{0,0}} d\, x_1^{0,0} + \frac{\partial w_{g_2}}{\partial x_2^{4,4}} d\, x_2^{4,4} + \frac{\partial w_{g}}{\partial x_2^{0,1}} d\, x_2^{0,1}$$

$$+ \frac{\partial w_{g_2}}{\partial x_3^{0,1}} d\, x_3^{0,1} + \frac{\partial w_{g_2}}{\partial x_3^{2,3}} d\, x_3^{2,3}$$

Therefore,

$$dw_{g_2} = (3\cdot x_1^{0,0}\, x_2^{0,1}\, x_3^{\overline{0,1}} + 3\cdot x_1^{\overline{0,0}}\, x_3^{\overline{0,1}})\, d\, x_1^{3,3} + 2\cdot x_2^{\overline{0,1}}\, d\, x_1^{0,0} + (2\cdot x_1^{0,0}\, x_3^{0,1}$$

$$+ 2\, x_1^{0,0}\, x_2^{0,1}\, x_3^{\overline{2,3}} + 2\cdot x_1^{1,2}\, x_3^{4,4} + 2\cdot x_1^{4,4}\, x_3^{4,4})\, d\, x_2^{4,4} + 4\cdot x_1^{0,0}\, d\, x_2^{0,1}$$

$$+ 3\cdot x_1^{3,3}\, d\, x_3^{0,1} + (2\cdot x_1^{1,2}\, x_2^{4,4} + 2\cdot x_1^{4,4}\, x_2^{4,4} + 2\cdot x_1^{3,3}\, x_2^{4,4}\, x_3^{0,1})\, d\, x_3^{2,3}$$

$$dw_{g_3} = \frac{\partial w_{g_3}}{\partial x_1^{3,3}}\, d\, x_1^{3,3} + \frac{\partial w_{g_3}}{\partial x_2^{4,4}}\, d\, x_2^{4,4} + \frac{\partial w_{g_3}}{\partial x_3^{0,1}}\, d\, x_3^{0,1} + \frac{\partial w_{g_3}}{\partial x_3^{2,3}}\, d\, x_3^{2,3} + \frac{\partial w_{g_3}}{\partial \; f_1}\, dw_{f_1}$$

$$= [(w_{f_1} + 2\cdot x_3^{\overline{0,1}} + 2\cdot x_2^{4,4}\, x_3^{2,3}) \oplus (w_{f_1} + 2\cdot x_2^{4,4}\, x_3^{2,3})]\, d\, x_1^{3,3}$$

$$+ [(w_{f_1} + 2\cdot x_1^{3,3}\, x_3^{\overline{0,1}} + 2\cdot x_3^{2,3}) \oplus (w_{f_1} + 2\cdot x_1^{3,3}\, x_3^{0,1})]\, d\, x_2^{4,4}$$

$$+ [(w_{f_1} + 2\cdot x_2^{4,4}\, x_3^{2,3}) \oplus (w_{f_1} + 2\cdot x_1^{3,3} + 2\cdot x_2^{4,4}\, x_3^{2,3})]\, d\, x_3^{0,1}$$

$$+ [(w_{f_1} + 2\cdot x_1^{3,3}\, x_3^{0,1} + 2\cdot x_2^{4,4}) \oplus (w_{f_1} + 2\cdot x_1^{3,3}\, x_3^{0,1})]\, d\, x_3^{2,3}$$

$$+ [(w_{f_1} + 2\cdot x_1^{3,3}\, x_3^{\overline{0,1}} + 2\cdot x_2^{4,4}\, x_3^{2,3}) \oplus (\overline{w}_f + 2\cdot x_1^{3,3}\, x_3^{\overline{0,1}}$$

$$+ 2\cdot x_2^{4,4}\, x_3^{2,3})]\, dw_{f_1}$$

$$dw_{h_1} = \frac{\partial w_{h_2}}{\partial x_1^{0,0}}\, d\, x_1^{0,0} + \frac{\partial w_{h_2}}{\partial x_1^{3,3}}\, d\, x_1^{3,3} + \frac{\partial w_{h_2}}{\partial x_2^{4,4}}\, d\, x_2^{4,4} + \frac{\partial w_{h_2}}{\partial x_3^{0,1}}\, d\, x_3^{0,1} + \frac{\partial w_{h_2}}{\partial x_3^{2,2}}\, d\, x_3^{2,2}$$

$$+ \frac{\partial w_{h_2}}{\partial x_3^{2,3}}\, d\, x_3^{2,3} + \frac{\partial w_{h_2}}{\partial w_{g_2}}\, dw_{g_2}$$

$$dw_{h_1} = [(w_{g_2} + 2\cdot x_1^{3,3}\, x_3^{0,1} + 1\cdot x_2^{4,4}\, x_3^{2,3} + 3\cdot x_3^{2,3}) \oplus (w_{g_2} + 2\cdot x_1^{3,3}\, x_3^{0,1}$$

$$+ 1\cdot x_2^{4,4}\, x_3^{2,2})]\, d\, x_1^{0,0} + [(w_{g_2} + 2\cdot x_3^{0,1} + 1\cdot x_2^{4,4}\, x_3^{2,3} + 3\cdot x_1^{0,0}\, x_3^{2,2})$$

$$\oplus\ (w_{g_2} + 1\cdot x_2^{4,4}\ x_3^{2,3} + 3\cdot x_1^{0,0}\ x_3^{2,2})]d\ x_1^{3,3} + [(w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1}$$

$$+ 1\cdot x_3^{2,3} + 3\cdot x_1^{0,0}\ x_3^{2,2}) \oplus (w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1} + 3\cdot x_1^{0,0}\ x_3^{2,2})]d\ x_2^{4,4}$$

$$+ [(w_{g_2} + 2\cdot x_1^{3,3} + 1\cdot x_2^{4,4}\ x_3^{2,3} + 3\cdot x_1^{0,0}\ x_3^{2,2}) \oplus (w_{g_2} + 1\cdot x_2^{4,4}\ x_3^{2,3}$$

$$+ 3\cdot x_1^{0,0}\ x_3^{2,2})]d\ x_3^{0,1} + [(w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1} + 1\cdot x_2^{4,4} + 3\cdot x_1^{0,0}\ x_3^{2,2})$$

$$\oplus\ (w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1} + 3\cdot x_1^{0,0}\ x_3^{2,2})]d\ x_3^{2,3} + [(w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1}$$

$$+ 1\cdot x_2^{4,4}\ x_3^{2,3} + 3\cdot x_1^{0,0}) \oplus (w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1} + 1\cdot x_2^{4,4}\ x_3^{2,2})]d\ x_3^{2,2}$$

$$+ [(w_{g_2} + 2\cdot x_1^{3,3}\ x_3^{0,1} + 1\cdot x_2^{4,4}\ x_3^{2,3} + 3\cdot x_1^{0,0}\ x_3^{2,2}) \oplus (\overline{w}_{g_2}$$

$$+ 2\cdot x_1^{3,3}\ x_3^{0,1} + 1\cdot x_2^{4,4}\ x_3^{2,3} + 3\cdot x_1^{0,0}\ x_3^{2,2})]dw_{g_2}$$

$$dz = \frac{\partial z}{\partial\ x_1^{0,0}}\ d\ x_1^{0,0} + \frac{\partial z}{\partial\ x_2^{0,0}}\ d\ x_2^{0,1} + \frac{\partial z}{\partial\ x_2^{4,4}}\ d\ x_2^{4,4} + \frac{\partial z}{\partial\ x_3^{2,3}}\ d\ x_3^{2,3} + \frac{\partial z}{\partial w_{f_6}}\ dw_{f_6}$$

$$+ \frac{\partial z}{\partial w_{g_3}}\ dw_{g_3} + \frac{\partial z}{\partial w_{h_1}}\ dw_h$$

$$dz = [(w_{h_1}\ w_{g_3} + w_{h_1}\ w_{f_6} + 1\cdot x_2^{4,4}\ x_3^{\overline{2,3}}\ w_{h_1} + 4\cdot x_2^{0,1}\ w_{h_1})$$

$$\oplus\ (w_{h_1}\ w_{g_3} + w_{h_1}\ w_{f_6} + 1\cdot x_2^{4,4}\ x_3^{\overline{2,3}}\ w_{h_1})]d\ x_1^{0,0} + [(w_{h_1}\ w_{g_3}$$

$$+ w_{h_1}\ w_{f_6} + 1\cdot x_2^{4,4}\ x_3^{\overline{2,3}}\ w_{h_1} + 4\cdot x_1^{0,0}\ w_{h_1}) \oplus (w_{h_1}\ w_{g_3}$$

$$+ w_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{\overline{2,3}}{x_3} w_{h_1}) d \overset{0,1}{x_2} + [(w_{h_1} w_{g_3} + w_{h_1} w_{f_6}$$

$$+ 1 \cdot \overset{\overline{2,3}}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1}) \oplus (w_{h_1} w_{g_3} + w_{h_1} w_{f_6}$$

$$+ 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})] d \overset{4,4}{x_2} + [(w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_h)$$

$$\oplus (w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})] d \overset{2,3}{x_3}$$

$$+ [(w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{\overline{2,3}}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})$$

$$\oplus (w_{h_1} w_{g_3} + w_{h_1} \overline{w}_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{2,3}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})] dw_{f_6}$$

$$+ [(w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{\overline{2,3}}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})$$

$$\oplus (w_{h_1} \overline{w}_{g_3} + w_{h_1} f_6 + 1 \cdot \overset{4,4}{x_2} \overset{\overline{2,3}}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})] dw_{g_2}$$

$$+ [(w_{h_1} w_{g_3} + w_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{2,3}{x_3} w_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} w_{h_1})$$

$$\oplus (\overline{w}_{h_1} w_{g_3} + \overline{w}_{h_1} w_{f_6} + 1 \cdot \overset{4,4}{x_2} \overset{\overline{2,3}}{x_3} \overline{w}_{h_1} + 4 \cdot \overset{0,0}{x_1} \overset{0,1}{x_2} \overline{w}_{h_1})] dw_{h_1}$$

Table 6.5.4 shows all static hazards that are present in the circuit of Figure 6.5.4. The types of static hazards are also indicated in Table 6.5.4.

TABLE 6.5.3

Static Hazard Table for Example 6.5.3

A $x_1x_2x_3$	B $x_1x_2x_3$	f(A)=f(B)	dw_{f_1}	dw_{f_6}	dw_{g_2}	dw_{g_3}	dw_{h_1}	w_{f_1}	w_{f_G}	w_{g_2}	w_{g_3}	w_{h_1}	dz	Type of Hazard
010	020	0	0	0	4	0	4	0	0		0		4	0-hazard
011	021	0	0	0	4	0	4	0	0		0		4	0-hazard
013	023	0	0	0	4	0	4	0	0		0		4	0-hazard
014	024	0	0	0	4	0	4	0	0		0		4	0-hazard
141	142	1	0	0	4	4	4	0	0				4	1-hazard
143	144	1	0	0	4	4	4	0	0				4	1-hazard
241	242	1	0	0	4	4	4	0	0				4	1-hazard
243	244	1	0	0	4	4	4	0	0				4	1-hazard
441	442	1	0	0	4	4	4	0	0				4	1-hazard
443	444	1	0	0	4	4	4	0	0				4	1-hazard
301	302	2	4	0	4	4	4		0				4	2-hazard
311	312	2	4	0	4	4	4		0				4	2-hazard
321	322	2	4	0	4	4	4		0				4	2-hazard
331	332	2	4	0	4	4	4		0				4	2-hazard
341	342	2	4	0	4	4	4		0				4	2-hazard

CHAPTER VII

STATIC HAZARD ELIMINATION IN ASYNCHRONOUS MULTI-LEVEL COMBINATIONAL CIRCUITS

7.1 Introduction

In Chapter VI, we have considered the problem of detecting the static hazard in asynchronous multi-level multi-valued combinational circuits using multi-valued digital calculus. From the practical point of view, it is not enough just to know the existance of the static hazard in any circuit without removing it. Therefore, this chapter is devoted to the problem of eliminating static hazard in asynchronous multi-level multi-valued combinational circuits.

The problem of static hazard elimination in two-valued asynchronous combinational circuits had first been investigated by Huffman [32] and Eichelberger [33] and then followed by many researchers [34],[36]. But most of the previous works on this subject [32],[33],[34],[36] were generally limited to two-level combinational circuits static hazard elimination. Since static 0 (static 1) hazards do not exist in two level AND-OR (OR-AND) asynchronous combinational circuits, the problem of static hazard elimination of two-level circuits is much simpler than that of multi-level circuits where both static 0 and static 1 hazards may exist. Thus, static hazard elimination of multi-level circuits is considered in this chapter. A procedure for static hazard elimination in multi-level multi-valued asynchronous combinational circuits is presented.

7.2 Static Hazard Elimination in Multi-Level Circuits

Static hazards in multi-level asynchronous combinational circuits can be eliminated since it depends on the realization of the circuits. It is found that static hazards can be eliminated in any multi-level multi-valued asynchronous combinational circuits by adding a redundant circuitry to the given realization. Redundant circuitry depends on the type of static hazard presented in the realization, i.e., whether the static hazard is 0-hazard or k-hazards, where k = 1,2,..., m-1. It is found that any multi-level multi-valued circuit can be free of all static k-hazards if the necessary implicants are included in the realization as stated in the following theorem.

<u>Theorem 7.2.1</u>

A multi-level multi-valued circuit will be free of all static k-hazards if and only if the realization contains all those k-implicants of ENF of the circuit that have at least two-adjacent cells that are not included in any other k-implicant.

<u>Proof</u>: To insure that static k-hazard is not present during

the transition from the input state A to the input state B, it is necessary that function f of the realization is equal to k during this transition. This means that there exists at least one product term in the ENFthat is equal to k during this transition. Therefore, k-implicant $k \cdot x_1^{b_1,b_1} x_2^{b_2,b_2} \cdots x_{n-1}^{b_{n-1},b_{n-1}}$ of $(b_1, b_2 \ldots, b_{n-1})$ is present in the realization either by itself or it is included in another k-implicant of the same k but it has less than n-1 input variables. Only those k-implicants of ENF that have at least two-adjacent cells that are not included in any other k-implicant are needed to be present in the realization because we are concerned only with single input change. This concludes the proof.

We conclude that redundant circuitry that is needed to eliminate all static k-hazard consists of all implicants that are not present in the ENF provided that these implicants must satisfy Theorem 7.2.1. This redundant circuitry will be called k-redundant circuitry because when it is ORed with the output of the original circuit, all static k-hazards are eliminated. Also f_k will be used to represent the output function of k-redundant circuitry.

It is found that static 0-hazard in any multi-level multi-valued circuit can be eliminated according to the following theorem.

Theorem 7.2.2

Static 0-hazard (if present) in any multi-level multi-valued circuit can be eliminated if an only if the function

f_0 is added to the original circuit by using AND gate, where f_0 is the complement of the summation of all minterms that correspond to those input states A and B (A and B are defined in Definition 6.1.1) that contained static 0-hazard.

Proof: The function f_0 is function of the input variables $(x_1,\ldots,x_n)$, i.e., $f_0 = f_0(x_1,\ldots,x_n)$. The values of $f_0(A)$ and $f_0(B)$ are always equal to zero for all those input states A and B that contain static 0-hazard. While the value of f_0 is equal to m-1 for all input states that do not contain static 0-hazard. Thus f_0 eliminates all static 0-hazards when it is ANDed with the output of the original circuit.

Redundant circuitry to eliminate all static 0-hazards will be called 0-redundant circuitry.

Circuits free of all static hazards can be obtained by first ORing k-redundant circuitry with the output of the given circuit to obtain a static k-hazard free circuit. Secondly, the static k-hazard free circuit obtained above is ANDed with 0-redundant circuitry. It is noticed that the same result can be obtained if the above procedure is reversed. Some multi-level circuits contain only one type of static hazard, i.e., either static k-hazard or static 0-hazard. If this is the case, then either k-redundant circuitry or 0-redundant circuitry need to be added to the given circuit, respectively. Adding k-redundant circuitry or 0-redundant circuitry or both to the original circuit does not change the truth table of the original circuit.

7.3 Procedure for Eliminating Static Hazards

A procedure for eliminating both static 0-hazard and static k-hazards in a multi-level multi-valued combinational circuit by first obtaining the equivalent normal form ENF of the given circuit is given below.

(1) Find ENF of the given multi-level multi-valued combinational circuit.

(2) Find all k-implicants that are not present in ENF obtained in (1) by using multi-valued map. These prime implicatns must be subjected to the condition of Theorem 7.2.1 to avoid more redundancy.

(3) Construct k-redundant circuitry from k-implicants obtained in (2). Add k-redundant circuitry to the original circuit by using OR gate.

(4) Construct 0-redundant circuitry by using Theorem 7.2.2. Add 0-redundant circuitry to the circuit obtained in (3) by using AND gate.

Example 7.3.1

Eliminate all static hazards present in the circuit of Figure 6.5.1 of Example 6.5.1. For convenience, this circuit is repeated in Figure 7.3.1.

By applying the above procedure, ENF of the circuit of Figure 7.3.1 is given in Equation (6.5.1). Equation (6.5.1) is rewritten in Equation (7.3.1) such that all literals subscripts are deleted.

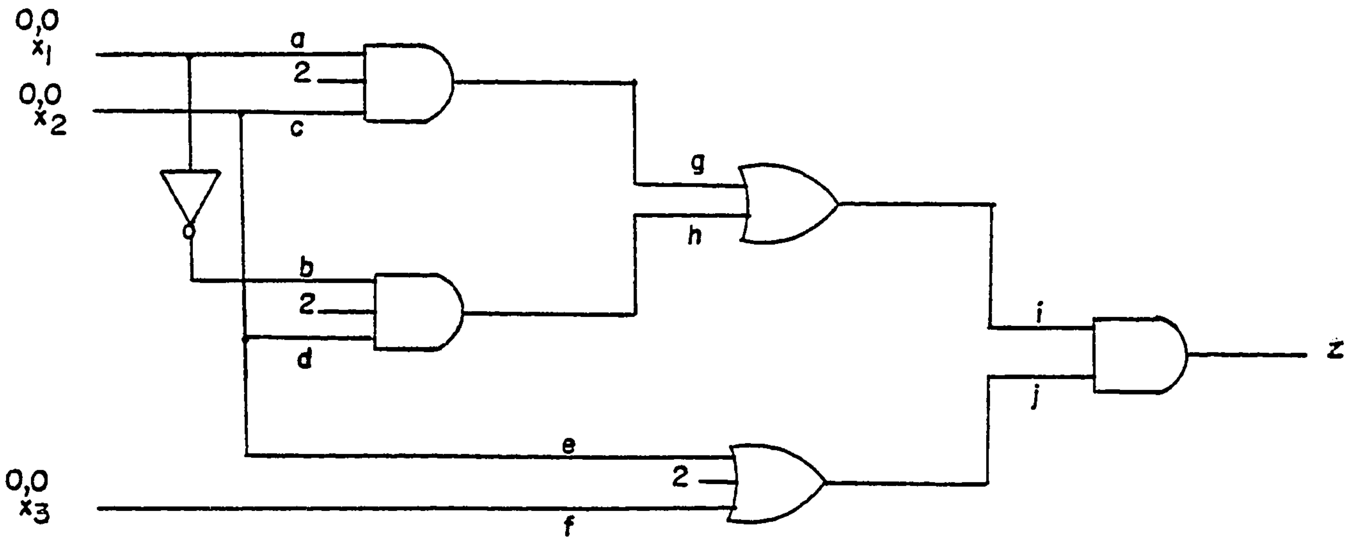


Figure 7.3.1: Circuit of Example 7.3.1.

$$z = 2\, x_1^{0,0}\; x_2^{0,0} + 2\cdot\, x_1^{0,0}\; x_2^{0,0}\; x_3^{0,0} + 2\cdot\, x_1^{\overline{0,0}}\; x_2^{0,0} + 2\cdot\, x_1^{0,0}\; x_2^{0,0}\; x_3^{0,0} \tag{7.3.1}$$

A 3-valued map of z of Equation (7.3.1) is shown in Figure (7.3.2).

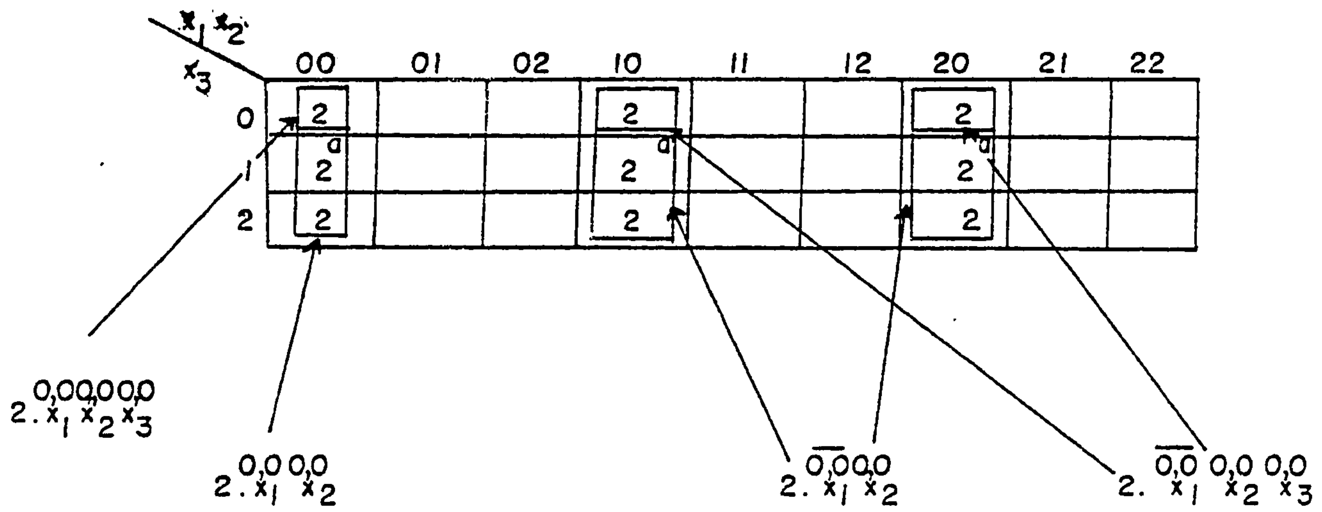


Figure 7.3.2: 3-Valued map of z of Equation (7.3.1).

The preceeding map shows that a-cubes can be combined to give 2-implicant $2\cdot \overset{0,0}{x_2}$. This implicant is not present in ENF of z in Equation (7.3.1). Thus, by adding this implicant to the circuit of Figure 7.3.1, all static 2-hazards are eliminated. k-redundant circuitry consists of only the implicant $2\cdot \overset{0,0}{x_2}$, i.e., $f_k = 2\cdot \overset{0,0}{x_2}$. Figure 7.3.3 shows static k-hazards free circuit of the circuit of Figure 7.3.1. Since the circuit of Figure 7.3.1 is free of all static 0-hazard as shown in Table 6.5.2, the circuit of Figure 7.3.3 is free of both static k-hazards and static 0-hazards.

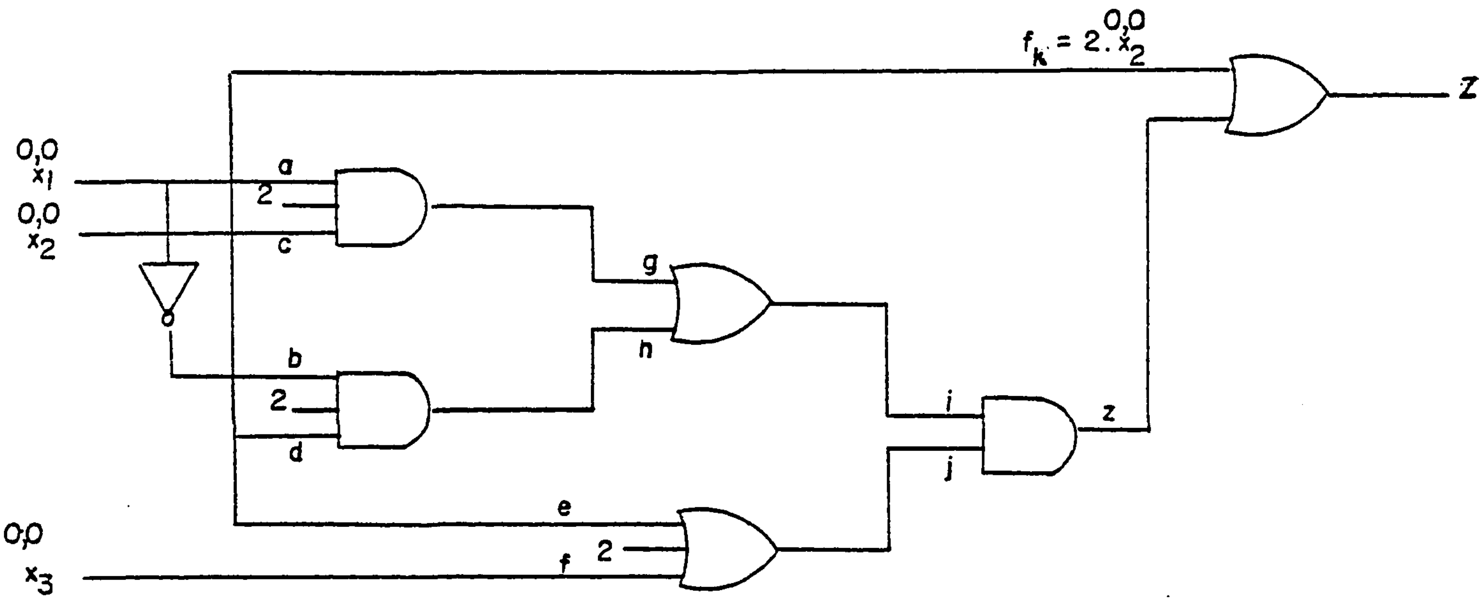


Figure 7.3.3: Static hazards free circuit of circuit of Figure 7.3.3.

Example 7.3.2

Eliminate all static hazards present in the circuit of Figure 6.5.4 of Example 6.5.3. For convenience, this circuit is repeated in Figure 7.3.4.

By applying static hazard elimination procedures, ENF of the circuit of Figure 7.3.4 is given by Equation (A.1.2). Equation (A.1.2) is rewritten in Equation (7.3.2) so that all literals subscripts are deleted and all terms that are equal to zero are also deleted.

$$\begin{aligned} z = {} & 2\cdot x_1^{3,3} x_3^{0,1} + 1\cdot x_1^{3,3} x_2^{4,4} x_3^{2,3} + 1\cdot x_2^{4,4} x_3^{2,3} + 2\cdot x_1^{0,0} x_2^{4,4} x_3^{2,2} \\ & + 1\cdot x_1^{3,3} x_2^{4,4} x_3^{0,1} + 1\cdot x_1^{0,0} x_2^{4,4} x_3^{2,2} + 3\cdot x_1^{0,0} x_2^{0,1} x_3^{2,2} + 3\cdot x_1^{0,0} x_3^{2,2} \\ & + 2\cdot x_1^{3,3} x_3^{\overline{0,1}} + 2\cdot x_1^{3,3} x_2^{4,4} x_3^{2,3} + 2\cdot x_1^{3,3} x_2^{4,4} x_3^{0,1} + 2\cdot x_1^{3,3} x_2^{4,4} x_3^{4,4} \\ & + 2\cdot x_1^{0,0} x_2^{4,4} x_3^{2,3} + 1\cdot x_1^{3,3} x_2^{4,4} x_3^{4,4} + 1\cdot x_2^{4,4} x_3^{\overline{2,3}} + 1\cdot x_1^{0,0} x_2^{4,4} x_3^{\overline{2,3}} \\ & + 3\cdot x_1^{0,0} x_2^{\overline{0,1}} x_3^{2,3} \end{aligned} \qquad (7.3.2)$$

Equation (7.3.2) can also be written as:

$$\begin{aligned} z = {} & 1\cdot[\, x_1^{3,3} x_2^{4,4} x_3^{2,3} + x_2^{4,4} x_3^{2,3} + x_1^{3,3} x_2^{4,4} x_3^{0,1} + x_1^{0,0} x_2^{4,4} x_3^{2,2} \\ & + x_1^{3,3} x_2^{4,4} x_3^{4,4} + x_2^{4,4} x_3^{\overline{2,3}} + x_1^{0,0} x_2^{4,4} x_3^{2,3}] + 2\cdot[\, x_1^{3,3} x_3^{0,1} \end{aligned}$$

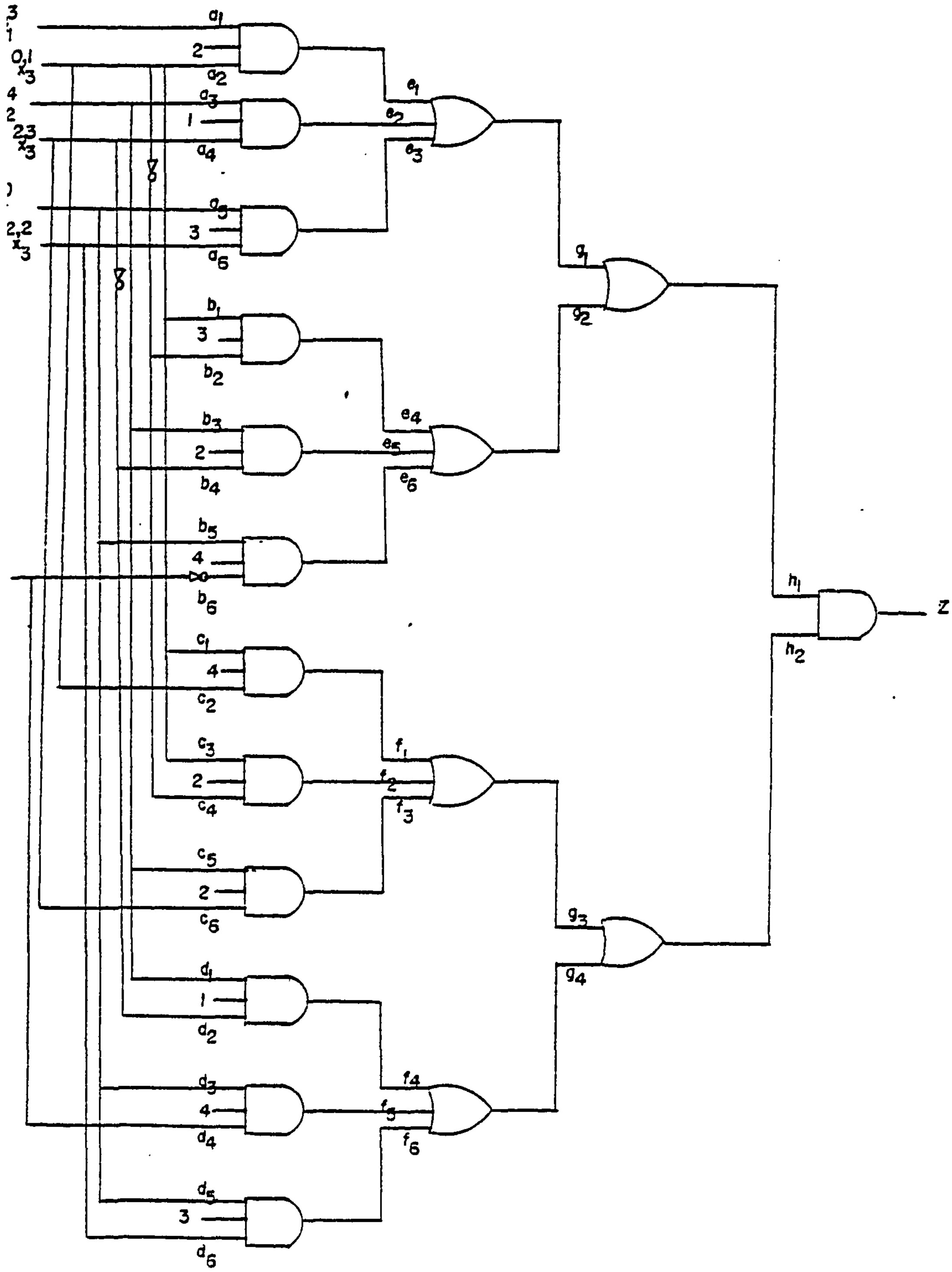


Figure 7.3.4: Circuit of Example 7.3.2

$$+ x_1^{0,0} x_2^{4,4} x_3^{2,2} + x_1^{3,3} x_3^{\overline{0,1}} + x_1^{3,3} x_2^{4,4} x_3^{2,3} + x_1^{3,3} x_2^{4,4} x_3^{0,1} +$$

$$+ x_1^{3,3} x_2^{4,4} x_3^{4,4} + x_1^{0,0} x_2^{4,4} x_3^{2,3}] + 3\cdot[x_1^{0,0} x_2^{0,1} x_3^{2,2} + x_1^{0,0} x_3^{2,2}$$

$$+ x_1^{0,0} x_2^{\overline{0,1}} x_3^{2,2}]$$

$$= z_1 + z_2 + z_3$$

where:

$$z_1 = 1\cdot[x_1^{3,3} x_2^{4,4} x_3^{2,3} + x_2^{4,4} x_3^{2,3} + x_1^{3,3} x_2^{4,4} x_3^{0,1} + x_1^{0,0} x_2^{4,4} x_3^{2,2}$$

$$+ x_1^{3,3} x_2^{4,4} x_3^{4,4} + x_2^{4,4} x_3^{\overline{2,3}} + x_1^{0,0} x_2^{4,4} x_3^{2,3}]$$

$$z_2 = 2\cdot[x_1^{3,3} x_3^{0,1} + x_1^{0,0} x_2^{4,4} x_3^{2,2} + x_1^{3,3} x_3^{\overline{0,1}} + x_1^{3,3} x_2^{4,4} x_3^{2,3}$$

$$+ x_1^{3,3} x_2^{4,4} x_3^{0,1} + x_1^{3,3} x_2^{4,4} x_3^{4,4} + x_1^{0,0} x_2^{4,4} x_3^{2,3}]$$

$$z_3 = 3\cdot[x_1^{0,0} x_2^{0,1} x_3^{2,2} + x_1^{0,0} x_3^{2,2} + x_1^{0,0} x_2^{\overline{0,1}} x_3^{2,2}]$$

Three 5-valued maps for z_1, z_2 and z_3 are shown in Figures (7.3.5), (7.3.6) and (7.3.7), respectively. All implicants of z_1, z_2 and z_3 are shown on these maps.

a-cubes of Figure 7.3.5 correspond to the implicant $1\cdot x_2^{4,4} x_3^{2,3}$ while b-cubes correspond to the implicant $1\cdot x_2^{4,4} x_3^{\overline{2,3}}$. The implicants $1\cdot x_1^{3,3} x_2^{4,4} x_3^{2,3}$, $1\cdot x_1^{0,0} x_2^{4,4} x_3^{2,2}$ and $1\cdot x_1^{0,0} x_2^{4,4} x_3^{2,3}$

of z_1 are covered by a-cubes whereas the implicants $1\cdot x_1^{3,3} x_2^{4,4} x_3^{0,1}$ and $x_1^{3,3} x_2^{4,4} x_3^{4,4}$ are covered by b-cubes. In other words, these implicants are not necessary to be shown on the map of z_1 in which their deletion does not affect the determination of the implicants that are stated in Theorem 7.2.1. From Figure 7.3.5, the implicant that is necessary to eliminate static 1-hazard is $1\cdot x_2^{4,4}$. The implicant $1\cdot x_2^{4,4}$ is covered by both cubes a and b.

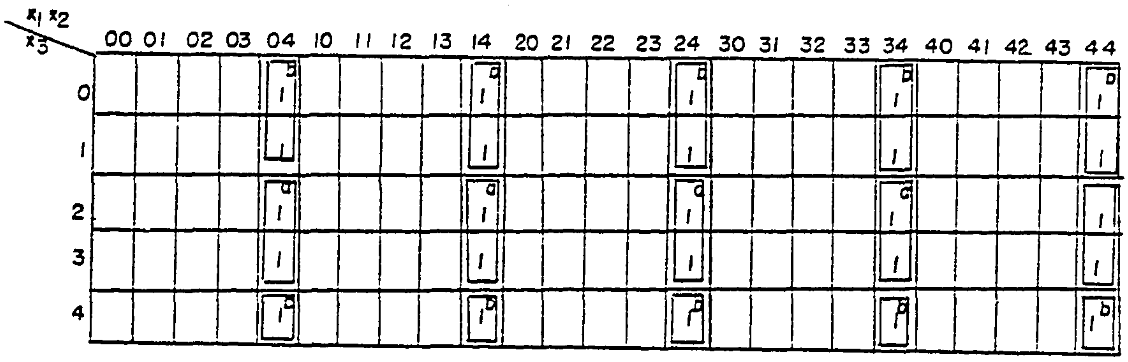


Figure 7.3.5: 5-Valued Map of z_1

c-cube, d-cube, and e-cube of Figure 7.3.6 correspond to the implicants $2\cdot x_1^{3,3} x_3^{0,1}$, $2\cdot x_1^{3,3} x_3^{\overline{0,1}}$ and $2\cdot x_1^{0,0} x_2^{4,4} x_3^{2,3}$ of z_2, respectively. The implicants $2\cdot x_1^{3,3} x_2^{4,4} x_3^{0,1}$; $2\cdot x_1^{3,3} x_2^{4,4} x_3^{2,3}$; $2\cdot x_1^{3,3} x_2^{4,4} x_3^{4,4}$; and $2\cdot x_1^{0,0} x_2^{4,4} x_3^{2,2}$ are covered by c-cube, d-cube and e-cube, respectively. From Figure 7.3.6, the implicant that is necessary to eliminate static 2-hazard is $2\cdot x_1^{3,3}$. The implicant $2\cdot x_1^{3,3}$ is covered by cubes c and d.

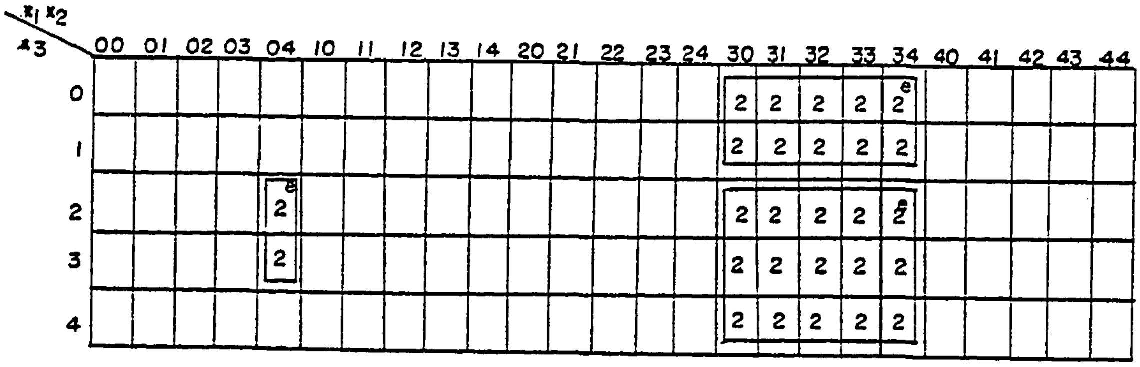


Figure 7.3.6: 5-Valued Map of z_2

f-cube of Figure 7.3.7 corresponds to the implicant $3\cdot x_1^{0,0}\ x_3^{2,2}$. The implicants $3\cdot x_1^{0,0}\ x_2^{0,1}\ x_3^{2,2}$ and $3\cdot x_1^{0,0}\ x_2^{\overline{0,1}}\ x_3^{2,2}$ are covered by f-cube. No implicant is needed for z_3 because all minterms are covered by the implicant $3\cdot x_1^{0,0}\ x_3^{2,2}$. The output function f_k of k-redundant circuitry is:

$$f_k = 1\cdot x_2^{4,4} + 2\cdot x_1^{3,3} \tag{7.3.3}$$

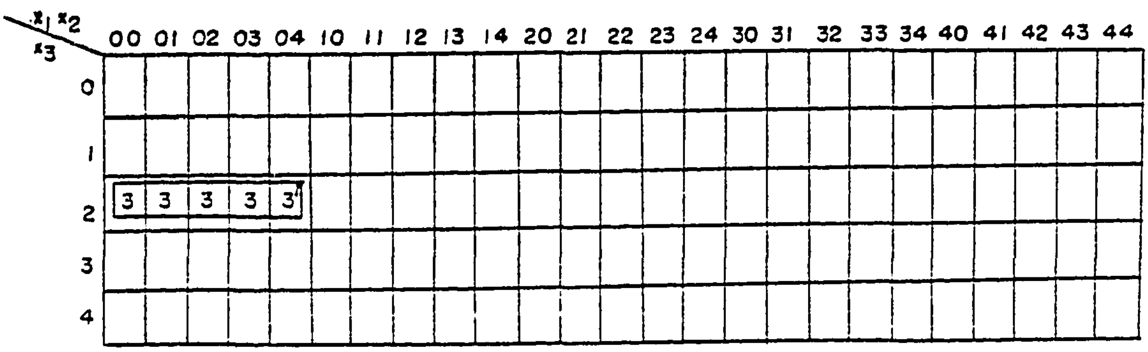


Figure 7.3.7: 5-Valued Map of z_3

From Table 6.5.4, the output function f_0 of 0-redundant circuitry is:

$$f_0 = (x_1^{0,0} x_2^{1,1} x_3^{0,0} + x_1^{0,0} x_2^{2,2} x_3^{0,0} + x_1^{0,0} x_2^{1,1} x_3^{1,1} + x_1^{0,0} x_2^{2,2} x_3^{1,1}$$
$$+ x_1^{0,0} x_2^{1,1} x_3^{3,3} + x_1^{0,0} x_2^{2,2} x_3^{3,3} + x_1^{0,0} x_2^{1,1} x_3^{4,4} + x_1^{0,0} x_2^{2,2} x_3^{4,4})'$$

function f_0 can be minimized as follows:

$$f_0 = [x_1^{0,0} x_3^{0,0} (x_2^{1,1} + x_2^{2,2}) + x_1^{0,0} x_3^{1,1} (x_2^{1,1} + x_2^{2,2}) + x_1^{0,0} x_3^{3,3}$$
$$(x_2^{1,1} + x_2^{2,2}) + x_1^{0,0} x_3^{4,4} (x_2^{1,1} + x_2^{2,2})]'$$
$$= [x_1^{0,0} x_2^{1,2} x_3^{0,0} + x_1^{0,0} x_2^{1,2} x_3^{1,1} + x_1^{0,0} x_2^{1,2} x_3^{3,3} + x_1^{0,0} x_2^{1,2} x_3^{4,4}]'$$
$$= [x_1^{0,0} x_2^{1,2} (x_3^{0,0} + x_3^{1,1} + x_3^{3,3} + x_3^{4,4})]'$$
$$= [x_1^{0,0} x_2^{1,2} (x_3^{0,0} + x_3^{3,4})]'$$

By using Equation (3.2.2), we get:

$$f_0 = (x_1^{0,0} x_2^{1,2} x_3^{\overline{2,2}})'$$
$$= (x_1^{\overline{0,0}} + x_2^{\overline{1,2}} + x_3^{2,2}) \qquad (7.3.4)$$

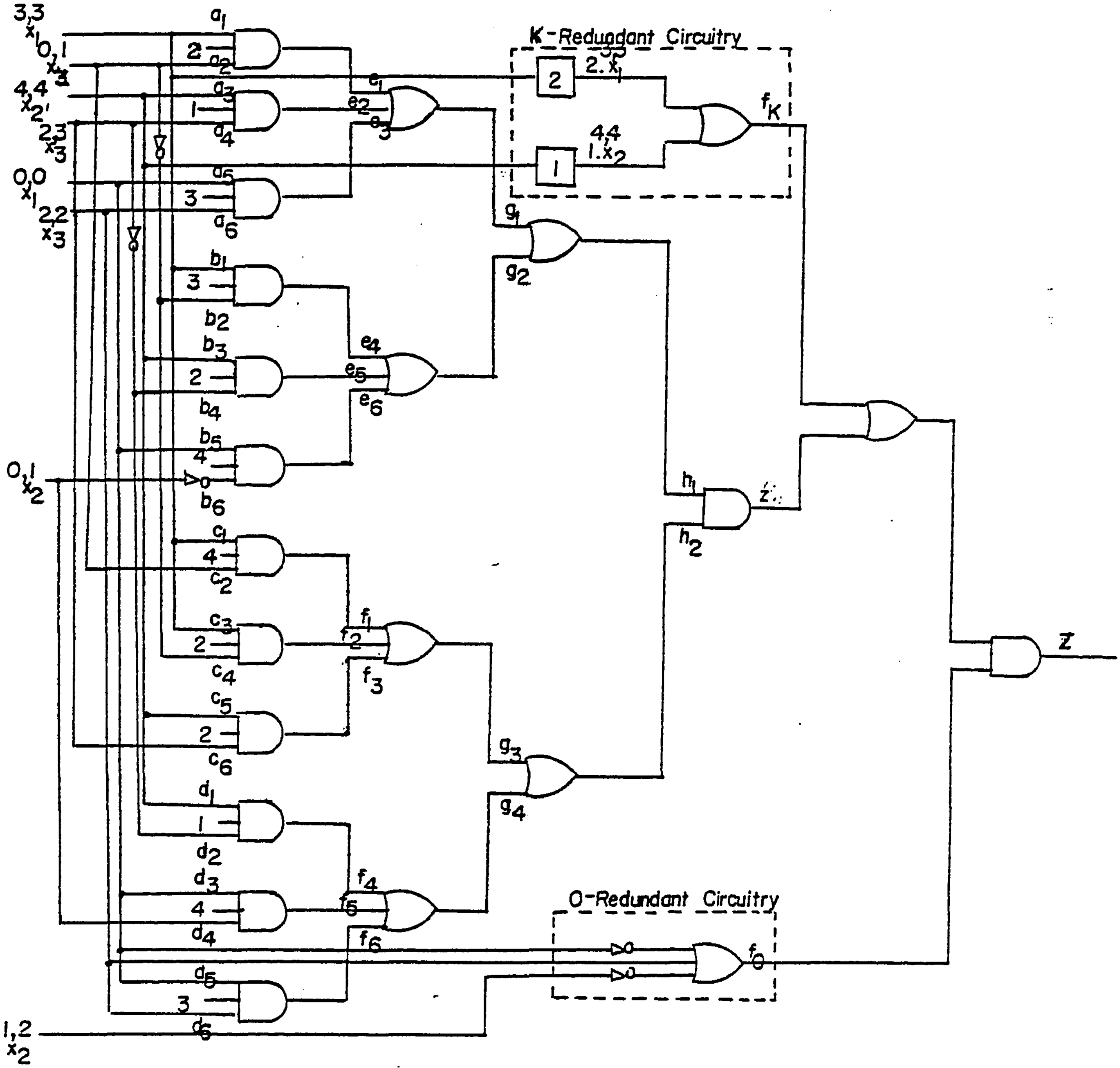


Figure 7.3.8: Static hazards free circuit of circuit of Figure 7.3.4.

By adding k-redundant circuitry and 0-redundant circuitry which are realized by the functions f_k and f_0 of Equations (7.3.3) and (7.3.4), respectively, to the circuit of Figure 7.3.4, static hazard free cricuit can be obtained as shown in Figure 7.3.8.

CHAPTER VIII

CONCLUSION

In this dissertation, two potentially implementable multi-valued algebras were given. Simple relations between these two algebras were presented and by using them one can obtain one system from the other. Multi-valued partial derivative and multi-valued total differential were introduced. Multi-valued partial derivative is used for test derivation for detecting single faults in multi-valued combinational circuits.

Two procedures for detecting and eliminating static hazards in asynchronous multi-level multi-valued combinational circuits have been presented. Since these procedures are derived for general multi-level multi-valued circuits, most previously published static hazard detection and elimination methods [32-38] for two-level two-valued circuit becomes special cases of these general procedures.

This work can be expanded to include test derivation for detecting multiple faults in multi-valued combinational circuit using multi-valued digital calculus. Dynamic hazard in multi-valued combinational circuits can be detected and eliminated by using multi-valued digital calculus. Fault detection and hazard detection of multi-valued sequential machines can also be investigated using multi-valued digital calculus techniques.

A.1 Equivalent Normal Form of Example 6.5.3

The following are the equations of the equivalent normal (ENF) of the circuit of Figure 6.5.4 of Example 6.5.3.

$$
\begin{aligned}
z = h_1\, h_2 &= (g_1 + g_2)_{h_1}\ (g_3 + g_4)_{h_2} \\
&= (g_{1_{h_1}} + g_{2_{h_1}})(g_{3_{h_2}} + g_{4_{h_2}}) \\
&= g_{1_{h_1}}\, g_{3_{h_2}} + g_{1_{h_1}}\, g_{4_{h_2}} + g_{2_{h_1}}\, g_{3_{h_2}} + g_{2_{h_1}}\, g_{4_{h_2}} \\
&= (e_1 + e_2 + e_3)_{g_1h_1}\ (f_1 + f_2 + f_3)_{g_3h_2} + (e_1 + e_2 \\
&\quad + e_3)_{g_1h_1}\ (f_4 + f_5 + f_6)_{g_4h_2} + (e_4 + e_5 + e_6)_{g_2h_1} \\
&\quad (f_1 + f_2 + f_3)_{g_3h_2} + (e_4 + e_5 + e_6)_{g_2h_1}\ (f_4 + f_5 \\
&\quad + f_6)_{g_4h_2} \\
&= (e_{1_{g_1h_1}} + e_{2_{g_1h_1}} + e_{3_{g_1h_1}})(f_{1_{g_3h_2}} + f_{2_{g_3h_2}} + f_{3_{g_3h_2}}) \\
&\quad + (e_{1_{g_1h_1}} + e_{2_{g_1h_1}} + e_{3_{g_1h_1}})(f_{4_{g_4h_2}} + f_{5_{g_4h_2}} + f_{6_{g_4h_2}}) \\
&\quad + (e_{4_{g_2h_1}} + e_{5_{g_2h_1}} + e_{6_{g_2h_1}})(f_{1_{g_3h_2}} + f_{2_{g_3h_2}} + f_{3_{g_3h_2}}) \\
&\quad + (e_{4_{g_2h_1}} + e_{5_{g_2h_1}} + e_{6_{g_2h_1}})(f_{4_{g_4h_2}} + f_{5_{g_4h_2}} + f_{6_{g_4h_2}}) \\
&= e_{1_{g_1h_1}}\, f_{1_{g_3h_2}} + e_{1_{g_1h_1}}\, f_{2_{g_3h_2}} + e_{1_{g_1h_1}}\, f_{3_{g_3h_2}} \\
&\quad + e_{2_{g_1h_1}}\, f_{1_{g_3h_2}} + e_{2_{g_1h_1}}\, f_{2_{g_3h_2}} + e_{2_{g_1h_1}}\, f_{3_{g_3h_2}}
\end{aligned}
$$

$$+ e_{3_{g_1 h_1}} f_{1_{g_3 h_2}} + e_{3_{g_1 h_1}} f_{2_{g_3 h_2}} + e_{3_{g_1 h_1}} f_{3_{g_3 h_2}}$$

$$+ e_{1_{g_1 h_1}} f_{4_{g_4 h_2}} + e_{1_{g_1 h_1}} f_{5_{g_4 h_2}} + e_{1_{g_1 h_1}} f_{6_{g_4 h_2}}$$

$$+ e_{2_{g_1 h_1}} f_{4_{g_4 h_2}} + e_{2_{g_1 h_1}} f_{5_{g_4 h_2}} + e_{2_{g_1 h_1}} f_{6_{g_4 h_2}}$$

$$+ e_{3_{g_1 h_1}} f_{4_{g_4 h_2}} + e_{3_{g_1 h_1}} f_{5_{g_4 h_2}} + e_{3_{g_1 h_1}} f_{6_{g_4 h_2}}$$

$$+ e_{4_{g_2 h_1}} f_{1_{g_3 h_2}} + e_{4_{g_2 h_1}} f_{2_{g_3 h_2}} + e_{4_{g_2 h_1}} f_{3_{g_3 h_2}}$$

$$+ e_{5_{g_2 h_1}} f_{1_{g_3 h_2}} + e_{5_{g_2 h_1}} f_{2_{g_3 h_2}} + e_{6_{g_2 h_1}} f_{3_{g_3 h_2}}$$

$$+ e_{6_{g_2 h_1}} f_{1_{g_3 h_2}} + e_{6_{g_2 h_1}} f_{2_{g_3 h_2}} + e_{6_{g_2 h_1}} f_{3_{g_3 h_2}}$$

$$+ e_{4_{g_2 h_1}} f_{4_{g_4 h_2}} + e_{4_{g_2 h_1}} f_{5_{g_4 h_2}} + e_{4_{g_2 h_1}} f_{6_{g_4 h_2}}$$

$$+ e_{5_{g_2 h_1}} f_{4_{g_4 h_2}} + e_{5_{g_2 h_1}} f_{5_{g_4 h_2}} + e_{5_{g_2 h_1}} f_{6_{g_4 h_2}}$$

$$+ e_{6_{g_2 h_1}} f_{4_{g_4 h_2}} + e_{6_{g_2 h_1}} f_{5_{g_4 h_2}} + e_{6_{g_2 h_1}} f_{6_{g_4 h_2}}$$

$$z = (2\cdot a_1 a_2)_{e_1 g_1 h_1} (4\cdot c_1 c_2)_{f_1 g_3 h_2} + (2\cdot a_1 a_2)_{e_1 g_1 h_1}$$

$$(2\cdot c_3 c_4)_{f_2 g_3 h_2} + (2\cdot a_1 a_2)_{e_1 g_1 h_1} (2\cdot c_5 c_6)_{f_3 g_3 h_2}$$

$$+ (1\cdot a_3 a_4)_{e_2 g_1 h_1} (4\cdot c_1 c_2)_{f_1 g_3 h_2} + (1\cdot a_3 a_4)_{e_2 g_1 h_1}$$

$$(2\cdot c_3 c_4)_{f_2 g_3 h_2} + (1\cdot a_3 a_4)_{e_2 g_1 h_1} (2\cdot c_5 c_6)_{f_3 g_3 h_2}$$

$$+ (3\cdot a_5a_6)e_3g_1h_1\ (4\cdot c_1c_2)f_1g_3h_2 + (3\cdot a_5a_6)e_3g_1h_1$$

$$(2\cdot c_3c_4)f_2g_3h_2 + (3\cdot a_5a_6)e_3g_1h_1\ (2\cdot c_5c_6)f_3g_3h_2$$

$$+ (2\cdot a_1a_2)e_1g_1h_1\ (1\cdot d_1d_2)f_4g_4h_4 + (2\cdot a_1a_2)e_1g_1h_1$$

$$(4\cdot d_3d_4)f_5g_4h_2 + (2\cdot a_1a_2)e_1g_1h_1\ (3\cdot d_5d_6)f_6g_4h_2$$

$$+ (1\cdot a_3a_4)e_2g_1h_1\ (1\cdot d_1d_2)f_4g_4h_1 + (1\cdot a_3a_4)e_2g_2h_2$$

$$(4\cdot d_3d_4)f_5g_4h_2 + (1\cdot a_3a_4)e_2g_2h_1\ (3\cdot d_5d_6)f_6g_4h_2$$

$$+ (3\cdot a_5a_6)e_3g_1h_1\ (1\cdot d_1d_2)f_4g_4h_1 + (3\cdot a_5a_6)e_3g_1h_1$$

$$(4\cdot d_3d_4)f_5g_4h_2 + (3\cdot a_5a_6)e_3g_1h_1\ (3\cdot d_5d_6)f_6g_4h_2$$

$$+ (3\cdot b_1b_2)e_4g_2h_1\ (4\cdot c_1c_2)f_1g_3h_2 + (3\cdot b_1b_2)e_4g_2h_1$$

$$(2\cdot c_3c_4)f_2g_3h_2 + (3\cdot b_1b_2)e_4g_2h_1\ (2\cdot c_5c_6)f_3g_3h_2$$

$$+ (2\cdot b_3b_4)e_5g_2h_1\ (4\cdot c_1c_2)f_1g_3h_2 + (2\cdot b_3b_4)e_5g_2h_1$$

$$(2\cdot c_3c_4)f_2g_3h_2 + (2\cdot b_3b_4)e_5g_2h_1\ (2\cdot c_5c_6)f_3g_3h_2$$

$$+ (4\cdot b_5b_6)e_6g_2h_1\ (4\cdot c_1c_2)f_1g_3h_2 + (4\cdot b_5b_6)e_6g_2h_1$$

$$(2\cdot c_3c_4)_{f_2g_3h_2} + (4\cdot b_5b_6)_{e_6g_2h_1}\ (2\cdot c_5c_6)_{f_3g_3h_2}$$

$$+ (3\cdot b_1b_2)_{e_4g_2h_1}\ (1\cdot d_1d_2)_{f_4g_4h_2} + (3\cdot b_1b_2)_{e_4g_2h_1}$$

$$(4\cdot d_3d_4)_{f_5g_4h_2} + (3\cdot b_1b_2)_{e_4g_2h_1}\ (3\cdot d_5d_6)_{f_6g_4h_2}$$

$$+ (2\cdot b_3b_4)_{e_5g_2h_1}\ (1\cdot d_1d_2)_{f_4g_4h_2} + (2\cdot b_3b_4)_{e_5g_2h_1}$$

$$(4\cdot d_3d_4)_{f_5g_4h_2} + (2\cdot b_3b_4)_{e_5g_2h_1}\ (3\cdot d_5d_6)_{f_6g_4h_2}$$

$$+ (4\cdot b_5b_6)_{e_6g_2h_1}\ (1\cdot d_1d_2)_{f_4g_4h_2}\ (4\cdot b_5b_6)_{e_6g_2h_1}$$

$$(4\cdot d_3d_4)_{f_5g_4h_2} + (4\cdot b_5b_6)_{e_6g_2h_1}\ (3\cdot d_5d_6)_{f_6g_4h_2}$$

$$= 2\ a_{1_{e_1g_1h_1}}\ a_{2_{e_1g_1h_1}}\ c_{1_{f_1g_3h_2}}\ c_{2_{f_1g_3h_2}} + 2\cdot a_{1_{e_1g_1h_1}}$$

$$a_{2_{e_1g_1h_1}}\ c_{3_{f_2g_3h_2}}\ c_{4_{f_2g_3h_2}} + 2\cdot a_{1_{e_1g_1h_1}}\ a_{2_{e_1g_1g_1}}$$

$$c_{5_{f_3g_3h_2}}\ c_{6_{f_3g_3h_2}} + 1\cdot a_{3_{e_2g_1h_1}}\ a_{4_{e_2g_1h_1}}\ c_{1_{f_1g_3h_2}}$$

$$c_{2_{f_1g_3h_2}} + 1\cdot a_{3_{e_2g_1h_1}}\ a_{4_{e_2g_1h_1}}\ c_{3_{f_2g_3h_2}}\ c_{4_{f_2g_3h_2}}$$

$$+ 1\cdot a_{3_{e_2g_1h_1}}\ a_{4_{e_2g_1h_1}}\ c_{5_{f_2g_3h_2}}\ c_{6_{f_3g_3h_2}} + 3\cdot a_{5_{e_3g_1h_1}}$$

$$a_{6_{e_3g_1h_1}}\ c_{1_{f_1g_3h_2}}\ c_{2_{f_1g_3h_2}} + 2\cdot a_{5_{e_3g_1h_1}}\ a_{6_{e_3g_1h_1}}$$

$$c_{3_{f_2g_3h_2}}\ c_{4_{f_2g_3h_2}} + 2\cdot a_{5_{e_3g_1h_1}}\ a_{6_{e_3g_1h_1}}\ c_{5_{f_3g_3h_2}}$$

$$c_{6_{f_3g_3h_2}} + 1\cdot a_{1_{e_1g_1h_1}}\ a_{2_{e_1g_1h_1}}\ d_{1_{f_4g_4h_2}}\ d_{2_{f_4g_4h_2}}$$

$$+ 2\cdot a_{1_{e_1g_1h_1}}\ a_{2_{e_1g_1h_1}}\ d_{3_{f_5g_4h_2}}\ d_{4_{f_5g_4h_2}} + 2\cdot a_{1_{e_1g_1h_1}}$$

$$a_{2_{e_1g_1h_1}}\ d_{5_{f_6g_4h_2}}\ d_{6_{f_6g_4h_2}} + 1\cdot a_{3_{e_2g_1h_1}}\ a_{4_{e_2g_1h_1}}$$

$$d_{1_{f_4g_4h_1}}\ d_{2_{f_4g_4h_1}} + 1\cdot a_{3_{e_2g_1h_1}}\ a_{4_{e_2g_1h_1}}\ d_{3_{f_5g_4h_2}}\ d_{4_{f_5g_4h_2}}$$

$$+ a_{3_{e_eg_2h_1}}\ a_{4_{e_2g_2h_1}}\ d_{5_{f_6g_4h_2}}\ d_{6_{f_6g_4h_2}} + 1\cdot a_{5_{e_3g_1h_1}}\ a_{6_{e_3g_1h_1}}$$

$$d_{1_{f_4g_4h_1}}\ d_{2_{f_4g_4h_1}} + 3\cdot a_{5_{e_3g_1h_1}}\ a_{6_{e_3g_1h_1}}\ d_{3_{f_5g_4h_2}}$$

$$d_{4_{f_5g_4h_2}} + 3\cdot a_{5_{e_3g_1h_1}}\ a_{6_{e_3g_1h_1}}\ d_{5_{f_6g_4h_2}}\ d_{6_{f_6g_4h_2}}$$

$$+ 3\cdot b_{1_{e_4g_2h_1}}\ b_{2_{e_4g_2h_1}}\ c_{1_{f_1g_3h_2}}\ c_{2_{f_1g_3h_2}} + 2\cdot b_{1_{e_4g_2h_1}}$$

$$b_{2_{e_4g_2h_1}}\ c_{3_{f_2g_3h_2}}\ c_{4_{f_2g_3h_2}} + 2\cdot b_{1_{e_4g_2h_1}}\ b_{2_{e_4g_2h_1}}$$

$$c_{5_{f_3g_3h_2}}\ c_{6_{f_3g_3h_2}} + 2\cdot b_{3_{e_5g_2h_1}}\ b_{4_{e_5g_2h_1}}\ c_{1_{f_1g_3h_2}}$$

$$c_{2_{f_1g_3h_2}} + 2\cdot b_{3_{e_5g_2h_1}}\ b_{4_{e_5g_2h_1}}\ c_{3_{f_2g_3h_2}}\ c_{4_{f_2g_3h_2}}$$

$$+ 2\cdot b_{3_{e_5g_2h_1}}\ b_{4_{e_5g_2h_1}}\ c_{5_{f_3g_3h_2}}\ c_{6_{f_3g_3h_2}} + 4\cdot b_{5_{e_6g_2h_1}}$$

$$
\begin{aligned}
& b_{6_{e_6 g_2 h_1}} \; c_{1_{f_1 g_3 h_2}} \; c_{2_{f_1 g_3 h_2}} + 2 \cdot b_{5_{e_6 g_2 h_1}} \; b_{6_{e_6 g_2 h_1}} \\
& c_{3_{f_2 g_3 h_2}} \; c_{4_{f_2 g_3 h_2}} + 2 \cdot b_{5_{e_6 g_2 h_1}} \; b_{6_{e_6 g_2 h_1}} \; c_{5_{f_3 g_3 h_2}} \\
& c_{6_{f_3 g_3 h_2}} + 1 \cdot b_{1_{e_4 g_2 h_1}} \; b_{2_{e_4 g_2 h_1}} \; d_{1_{f_4 g_4 h_2}} \; d_{2_{f_4 g_4 h_2}} \\
& + 3 \cdot b_{1_{e_4 g_2 h_1}} \; b_{2_{e_4 g_2 h_1}} \; d_{3_{f_5 g_4 h_2}} \; d_{4_{f_5 g_4 h_2}} + 3 \cdot b_{1_{e_4 g_2 h_1}} \\
& b_{2_{e_4 g_2 h_1}} \; d_{5_{f_6 g_4 h_2}} \; d_{6_{f_6 g_4 h_2}} + 1 \cdot b_{3_{e_5 g_2 h_1}} \; b_{4_{e_5 g_2 h_1}} \\
& d_{1_{f_4 g_4 h_2}} \; d_{2_{f_4 g_4 h_2}} + 2 \cdot b_{3_{e_5 g_2 h_1}} \; b_{4_{e_5 g_2 h_1}} \; d_{3_{f_5 g_4 h_2}} \\
& d_{4_{f_5 g_4 h_2}} + 2 \cdot b_{3_{e_5 g_2 h_1}} \; b_{4_{e_5 g_2 h_1}} \; d_{5_{f_6 g_4 h_2}} \; d_{6_{f_6 g_4 h_2}} \\
& + 1 \cdot b_{5_{e_6 g_2 h_1}} \; b_{6_{e_6 g_2 h_1}} \; d_{1_{f_4 g_4 h_2}} \; d_{2_{f_4 g_4 h_2}} + 4 \cdot b_{5_{e_6 g_2 h_1}} \\
& b_{6_{e_6 g_2 h_1}} \; d_{3_{f_5 g_4 h_2}} \; d_{4_{f_5 g_4 h_2}} + 3 \cdot b_{5_{e_6 g_2 h_1}} \; b_{6_{e_6 g_2 h_1}} \\
& d_{5_{f_6 g_4 h_2}} \; d_{6_{f_6 g_4 h_2}} \qquad \text{(A.1.1)}
\end{aligned}
$$

By substituting the input variable of the circuit of Figure 6.5.4 into Equation (A.1.1), we obtain:

$$
z = 2 \cdot x_{1_{a_1 e_1 g_1 h_1}}^{3,3} \; x_{3_{a_2 e_1 g_1 h_1}}^{0,0} \; x_{1_{c_1 f_1 g_3 h_2}}^{3,3} \; x_{3_{c_2 f_1 g_3 h_2}}^{0,1}
$$

$$+\ 2\cdot\ x^{3,3}_{1_{a_1e_1g_1h_1}}\ x^{0,1}_{3_{a_2e_1g_1h_2}}\ x^{3,3}_{1_{c_3f_2g_3h_2}}\ x^{\overline{0,1}}_{3_{c_4f_2g_3h_2}}$$

$$+\ 2\cdot\ x^{3,3}_{1_{a_1e_1g_1h_1}}\ x^{0,1}_{3_{a_2e_1g_1h_1}}\ x^{4,4}_{2_{c_5f_3g_3h_2}}\ x^{2,3}_{3_{c_6f_3g_3h_2}}$$

$$+\ 1\cdot\ x^{4,4}_{2_{a_3e_2g_1h_1}}\ x^{2,3}_{3_{a_4e_2g_1h_1}}\ x^{3,3}_{1_{c_1f_1g_2h_2}}\ x^{0,1}_{3_{c_2f_1g_3h_2}}$$

$$+\ 1\cdot\ x^{4,4}_{2_{a_3e_2g_1h_1}}\ x^{2,3}_{3_{a_4e_2g_1h_1}}\ x^{3,3}_{1_{c_3f_2g_3h_2}}\ x^{\overline{0,1}}_{3_{c_4f_2g_3h_2}}$$

$$+\ 1\cdot\ x^{4,4}_{2_{a_3e_2g_1h_1}}\ x^{2,3}_{3_{a_4e_2g_1h_1}}\ x^{4,4}_{2_{c_5f_3g_3h_2}}\ x^{2,3}_{3_{c_6f_3g_3h_2}}$$

$$+\ 3\cdot\ x^{0,0}_{1_{a_5e_3g_1h_1}}\ x^{2,2}_{3_{a_6e_3g_1h_1}}\ x^{3,3}_{1_{c_1f_1g_3h_1}}\ x^{0,1}_{3_{c_2f_1g_3h_2}}$$

$$+\ 2\cdot\ x^{0,0}_{1_{a_5e_3g_1h_1}}\ x^{2,2}_{3_{a_6e_3g_1h_1}}\ x^{3,3}_{1_{c_3f_2g_3h_2}}\ x^{\overline{0,1}}_{3_{c_4f_2g_3h_2}}$$

$$+\ 2\cdot\ x^{0,0}_{1_{a_5e_3g_1h_1}}\ x^{2,2}_{3_{a_6e_3g_1h_1}}\ x^{4,4}_{2_{c_5f_3g_3h_2}}\ x^{2,3}_{3_{c_6f_3g_3h_2}}$$

$$+\ 1\cdot\ x^{3,3}_{1_{a_1e_1g_1h_1}}\ x^{0,1}_{3_{a_2e_1g_1h_1}}\ x^{4,4}_{2_{d_1f_4g_4h_2}}\ x^{\overline{2,3}}_{3_{d_2f_4g_4h_2}}$$

$$+\ 2\cdot\ x^{3,3}_{1_{a_1e_1g_1h_1}}\ x^{0,1}_{3_{a_2e_1g_1h_1}}\ x^{0,0}_{1_{d_3f_5g_4h_2}}\ x^{0,1}_{2_{d_4f_5g_4h_2}}$$

$$+\ 2\cdot\ x^{3,3}_{1_{a_1e_1g_1h_1}}\ x^{0,1}_{3_{a_2e_1g_1h_1}}\ x^{0,0}_{1_{d_5f_6g_4h_2}}\ x^{2,2}_{3_{d_6f_6g_4h_2}}$$

$$+\ 1\cdot\ x^{4,4}_{2_{a_3e_2g_1h_1}}\ x^{2,3}_{3_{a_4e_2g_1h_1}}\ x^{4,4}_{2_{d_1f_4g_4h_1}}\ x^{\overline{2,3}}_{3_{d_2f_4g_4h_1}}$$

$$+\ 1\cdot\ x^{4,4}_{2_{a_3e_2g_1h_1}}\ x^{2,3}_{3_{a_4e_2g_1h_1}}\ x^{0,0}_{1_{d_3f_5g_4h_2}}\ x^{0,1}_{2_{d_4f_5g_4h_2}}$$

$$+\,1\cdot x_{2_{a_3e_2g_1h_1}}^{4,4}\; x_{3_{a_4e_2g_1h_1}}^{2,3}\; x_{1_{d_5f_6g_4h_2}}^{0,0}\; x_{3_{d_6f_6g_4h_2}}^{2,2}$$

$$+\,1\cdot x_{1_{a_5e_3g_1h_1}}^{0,0}\; x_{3_{a_6e_3g_1h_1}}^{2,2}\; x_{2_{d_1f_4g_4h_1}}^{4,4}\; x_{3_{d_2f_4g_4h_1}}^{\overline{2,3}}$$

$$+\,3\cdot x_{1_{a_5e_3g_1h_1}}^{0,0}\; x_{3_{a_6e_3g_1h_1}}^{2,2}\; x_{1_{d_3f_5g_4h_2}}^{0,0}\; x_{2_{d_4f_5g_4h_2}}^{0,1}$$

$$+\,3\cdot x_{1_{a_5e_3g_1h_1}}^{0,0}\; x_{3_{a_6e_3g_1h_1}}^{2,2}\; x_{1_{d_5f_6g_4h_2}}^{0,0}\; x_{3_{d_6f_6g_4h_2}}^{2,2}$$

$$+\,3\cdot x_{1_{b_1e_4g_2h_1}}^{3,3}\; x_{3_{b_2e_4g_2h_1}}^{\overline{0,1}}\; x_{1_{c_1f_1g_3h_2}}^{3,3}\; x_{3_{c_2f_1g_3h_2}}^{0,1}$$

$$+\,2\cdot x_{1_{b_1e_4g_2h_1}}^{3,3}\; x_{3_{b_2e_4g_2h_1}}^{\overline{0,0}}\; x_{1_{c_3f_2g_3h_2}}^{3,3}\; x_{3_{c_4f_2g_3h_2}}^{\overline{0,1}}$$

$$+\,2\cdot x_{1_{b_1e_4g_2h_1}}^{3,3}\; x_{3_{b_2e_4g_2h_1}}^{\overline{0,1}}\; x_{2_{c_5f_3g_3h_2}}^{4,4}\; x_{3_{c_6f_3g_3h_2}}^{2,3}$$

$$+\,2\cdot x_{2_{b_3e_5g_2h_1}}^{4,4}\; x_{3_{b_4e_5g_2h_1}}^{\overline{2,3}}\; x_{1_{c_1f_1g_3h_2}}^{3,3}\; x_{3_{c_2f_1g_3h_2}}^{0,1}$$

$$+\,2\cdot x_{2_{b_3e_5g_2h_1}}^{4,4}\; x_{3_{b_4e_5g_2h_1}}^{\overline{2,3}}\; x_{1_{c_3f_2g_3h_2}}^{3,3}\; x_{3_{c_4f_2g_3h_2}}^{\overline{0,1}}$$

$$+\,2\cdot x_{2_{b_3e_5g_2h_1}}^{4,4}\; x_{3_{b_4e_5g_2h_1}}^{\overline{2,3}}\; x_{2_{c_5f_3g_3h_2}}^{4,4}\; x_{3_{c_6f_3g_3h_2}}^{2,3}$$

$$+\,4\cdot x_{1_{b_5e_6g_2h_1}}^{0,0}\; x_{2_{b_6e_6g_2h_1}}^{\overline{0,1}}\; x_{1_{c_1f_1g_3h_2}}^{3,3}\; x_{3_{c_2f_1g_3h_2}}^{0,1}$$

$$+\,2\cdot x_{1_{b_5e_6g_2h_1}}^{0,0}\; x_{2_{b_6e_6g_2h_1}}^{\overline{0,1}}\; x_{1_{c_3f_2g_3h_2}}^{3,3}\; x_{3_{c_4f_2g_3h_2}}^{\overline{0,1}}$$

$$+\,2\cdot x_{1_{b_5e_6g_2h_1}}^{0,0}\; x_{2_{b_6e_6g_2h_1}}^{\overline{0,1}}\; x_{2_{c_5f_3g_3h_2}}^{4,4}\; x_{2_{c_5f_3g_3h_2}}^{2,3}$$

$$
\begin{aligned}
&+ 1\cdot\, x_{1\,b_1e_4g_2h_1}^{3,3}\; x_{3\,b_2e_4g_2h_1}^{\overline{0,1}}\; x_{2\,d_1f_4g_4h_2}^{4,4}\; x_{3\,d_2f_4g_4h_2}^{\overline{2,3}}\\
&+ 3\cdot\, x_{1\,b_1e_4g_2h_1}^{3,3}\; x_{3\,b_2e_4g_2h_1}^{\overline{0,1}}\; x_{1\,d_3f_4g_4h_2}^{0,0}\; x_{2\,d_4f_5g_4h_2}^{0,1}\\
&+ 3\cdot\, x_{1\,b_1e_4g_2h_1}^{3,3}\; x_{3\,b_2e_4g_2h_1}^{\overline{0,1}}\; x_{1\,d_5f_6g_4h_2}^{0,0}\; x_{3\,d_6f_6g_4h_2}^{2,2}\\
&+ 1\cdot\, x_{2\,b_3e_5g_2h_1}^{4,4}\; x_{3\,b_4e_5g_2h_1}^{\overline{2,3}}\; x_{2\,d_1f_4g_4h_2}^{4,4}\; x_{3\,d_2f_4g_4h_2}^{\overline{2,3}}\\
&+ 2\cdot\, x_{2\,b_3e_5g_2h_1}^{4,4}\; x_{3\,b_4e_5g_2h_1}^{\overline{2,3}}\; x_{1\,d_3f_5g_4h_2}^{0,0}\; x_{2\,d_2f_5g_4h_2}^{0,1}\\
&+ 2\cdot\, x_{2\,b_3e_5g_2h_1}^{4,4}\; x_{3\,b_4e_5g_2h_1}^{\overline{2,3}}\; x_{1\,d_5f_6g_4h_2}^{0,0}\; x_{3\,d_6f_6g_4h_2}^{2,2}\\
&+ 1\cdot\, x_{1\,b_5e_6g_2h_1}^{0,0}\; x_{2\,b_6e_6g_2h_1}^{\overline{0,1}}\; x_{2\,d_1f_4g_4h_2}^{4,4}\; x_{3\,d_2f_4g_4h_2}^{\overline{2,3}}\\
&+ 4\cdot\, x_{1\,b_5e_6g_2h_1}^{0,0}\; x_{2\,b_6e_6g_2h_1}^{\overline{0,1}}\; x_{1\,d_3f_5g_4h_2}^{0,0}\; x_{2\,d_4f_5g_4h_2}^{0,1}\\
&+ 3\cdot\, x_{1\,b_5e_6g_2h_1}^{0,0}\; x_{2\,b_6e_6g_2h_1}^{\overline{0,1}}\; x_{1\,d_5f_6g_4h_2}^{0,0}\; x_{3\,d_6f_6g_4h_2}^{2,2}
\end{aligned}
\qquad \text{(A.1.2)}
$$

A.2 Minimal Sets of Parallel Paths of Example 6.5.3

The sets of longest parallel paths for the input variables $x_1^{3,3}$, $x_3^{0,1}$, $x_2^{4,4}$, $x_3^{2,3}$, $x_1^{0,0}$, $x_3^{2,2}$ and $x_2^{0,1}$ that can be obtained from Equation (A.1.2) are respectively:

$[(a_1,e_1,g_1,h_1),(c_1,f_1,g_3,h_2),(c_3,f_2,g_3,h_2),(b_1,e_4,g_2,h_1)]$,

$[(a_2,e_1,g_1,h_1),(c_2,f_1,g_3,h_2),(c_4,f_2,g_3,h_2),(b_2,e_4,g_2,h_1)]$,

$[(a_3,e_2,g_1,h_1),(c_5,f_3,g_3,h_2),(d_1,f_4,g_4,h_2),(b_3,e_5,g_2,h_1)]$,

$[(c_6,f_3,g_3,h_2),(a_4,e_3,g_1,h_1),(d_2,f_4,g_4,h_2),(b_4,e_5,g_2,h_1)]$,

$[(a_5,e_3,g_1,h_1),(d_3,f_5,g_4,h_2),(d_5,f_6,g_4,h_2),(b_5,e_6,g_2,h_1)]$,

$[(a_6,e_3,g_1,h_1),(d_6,f_6,g_4,h_2)]$, and $[(d_4,f_5,g_4,h_2),(b_6,e_6,g_2,h_1)]$

The searching trees for the first five sets of longest parallel paths are shown in Figures (A.2.1), (A.2.2), (A.2.3) (A.2.4), and (A.2.5), respectively. The last two sets of longest parallel paths and all sets of parallel paths that can be obtained from the searching trees of Figures (A.2.1), (A.2.2), (A.2.3), (A.2.4) and (A.2.5) are listed below. A number is assigned to each set to represent this set.

(1) (a_6,e_3,g_1,h_1) and (d_6,f_6,g_4,h_2)

(2) (d_4,f_5,g_4,h_2) and (b_6,e_6,g_2,h_1)

(3) $(a_1,e_1,g_1,h_1),(c_1,f_1,g_3,h_2),(c_3,f_2,g_3,h_2)$ and (b_1,e_4,g_2,h_1)

(4) $(a_1, e_1, g_1, h_1), (c_1, f_1, g_3, h_2)$, and (b_1, e_4, g_2, h_1)

(5) $(a_1, e_1, g_1, h_1), (e_1, f_1, g_3, h_2)$ and (c_3, f_2, g_3, h_2)

(6) $(c_1, f_1, g_3, h_2), (c_3, f_2, g_3, h_2)$ and (b_1, e_4, g_2, h_1)

(7) $(a_1, e_1, g_1, h_1), (c_3, f_2, g_3, h_2)$ and (b_1, e_4, g_2, h_1)

(8) (a_1, e_1, g_1, h_1) and (c_1, f_1, g_3, h_2)

(9) (a_1, e_1, g_1, h_1) and (c_3, f_2, g_3, h_2)

(10) (a_1, e_1, g_1, h_1) and (b_1, e_4, g_2, h_1)

(11) (c_1, f_1, g_3, h_2) and (c_3, f_2, g_3, h_2)

(12) (c_1, f_1, g_3, h_2) and (b_1, e_4, g_2, h_1)

(13) (c_3, f_2, g_3, h_2) and (b_1, e_4, g_2, h_1)

(14) (a_1, e_1, g_1) and (b_1, e_4, g_2)

(15) (c_1, f_1, g_3) and (c_3, f_2, g_3)

(16) (c_1,f_1) and (c_3,f_2)

(17) (a_2,e_1,g_1,h_1), (e_2,f_1,g_3,h_2), (c_4,f_2,g_3,h_2) and (b_2,e_4,g_2,h_1)

(18) (a_2,e_1,g_1,h_1), (c_4,f_2,g_3,h_2) and (b_2,e_4,g_2,h_1)

(19) (c_2,f_1,g_3,h_2), (c_4,f_2,g_3,h_2) and (b_2,e_4,g_2,h_1)

(20) (a_2,e_1,g_1,h_1), (c_2,f_1,g_3,h_2) and (c_4,f_2,g_3,h_2)

(21) (a_2,e_1,g_1,h_1), (c_2,f_1,g_3,h_2) and (b_2,e_4,g_2,h_1)

(22) (a_2,e_1,g_1,h_1) and (c_2,f_1,g_3,h_2)

(23) (a_2,e_1,g_1,h_1) and (c_4,f_2,g_3,h_2)

(24) (a_2,e_1,g_1,h_1) and (b_2,e_4,g_2,h_1)

(25) (c_2,f_1,g_3,h_2) and (c_4,f_2,g_3,h_2)

(26) (c_2,f_1,g_3,h_2) and (b_2,e_4,g_2,h_1)

(27) (c_4,f_2,g_3,h_2) and (b_2,e_4,g_2,h_1)

(28) (a_2,e_1,g_1) and (b_2,e_4,g_2)

(29) (c_2, f_1, g_3) and (c_4, f_2, g_3)

(30) (c_2, f_1) and (c_4, f_2)

(31) (a_3, e_2, g_1, h_1), (c_5, f_3, g_3, h_2), (d_1, f_4, g_4, h_2) and (b_3, e_5, g_2, h_1)

(32) (a_3, e_2, g_1, h_1), (d_1, f_4, g_4, h_2) and (b_3, e_5, g_2, h_1)

(33) (c_5, f_3, g_3, h_2), (d_1, f_4, g_4, h_2) and (b_3, e_5, g_2, h_1)

(34) (a_3, e_2, g_1, h_1), (c_5, f_3, g_3, h_2) and (d_1, f_4, g_4, h_2)

(35) (a_3, e_2, g_1, h_1), (c_5, f_3, g_3, h_2) and (b_3, e_5, g_2, h_1)

(36) (a_3, e_2, g_1, h_1) and (c_5, f_3, g_3, h_2)

(37) (a_3, e_2, g_1, h_1) and (d_1, f_4, g_4, h_2)

(38) (a_3, e_2, g_1, h_1) and (b_3, e_5, g_2, h_1)

(39) (c_5, f_3, g_3, h_2) and (d_1, f_4, g_4, h_2)

(40) (c_5, f_3, g_3, h_2) and (b_3, e_5, g_2, h_1)

(41) (d_1, f_4, g_4, h_2) and (b_3, e_5, g_2, h_1)

(42) (a_3, e_2, g_1) and (b_3, e_5, g_2)

(43) (c_5, f_3, g_3) and (d_1, f_4, g_4)

(44) (c_6, f_3, g_3, h_2), (a_4, e_2, g_1, h_1), (d_2, f_4, g_4, h_2) and (b_4, e_5, g_2, h_1)

(45) (c_6, f_3, g_3, h_2), (d_2, f_4, g_4, h_2) and (b_4, e_5, g_2, h_1)

(46) (a_4, e_2, g_1, h_1), (d_2, f_4, g_4, h_2) and (b_4, e_5, g_2, h_1)

(47) (c_6, f_3, g_3, h_2), (a_4, e_2, g_1, h_1) and (d_2, f_4, g_4, h_2)

(48) (c_6, f_3, g_3, h_2), (a_4, e_2, g_1, h_1) and (b_4, e_5, g_2, h_1)

(49) (c_6, f_3, g_3, h_2) and (a_4, e_2, g_1, h_1)

(50) (c_6, f_3, g_3, h_2) and (d_2, f_4, g_4, h_2)

(51) (c_6, f_3, g_3, h_2) and (b_4, e_6, g_2, h_1)

(52) (a_4, e_2, g_1, h_1) and (d_2, f_4, g_4, h_2)

(53) (a_4, e_2, g_1, h_1) and (b_4, e_5, g_2, h_1)

(54) (d_2, f_4, g_4, h_2) and (b_4, e_5, g_2, h_1)

(55) (c_6, f_3, g_3) and (d_2, f_4, g_4)

(56) (a_4, e_2, g_1) and (b_4, e_5, g_2)

(57) $(a_5, e_3, g_1, h_1), (d_3, f_5, g_4, h_2), (d_5, f_6, g_4, h_2)$ and (b_5, e_6, g_2, h_1)

(58) $(a_5, e_3, g_1, h_1), (d_5, f_6, g_4, h_2)$ and (b_5, e_6, g_2, h_1)

(59) $(d_3, f_5, g_4, h_2), (d_5, f_6, g_4, h_2)$ and (b_5, e_6, g_2, h_1)

(60) $(a_5, e_3, g_1, h_1), (d_3, f_5, g_4, h_2)$ and (d_5, f_6, g_4, h_2)

(61) $(a_5, e_3, g_1, h_1), (d_3, f_5, g_4, h_2)$ and (b_5, e_6, g_2, h_1)

(62) (a_5, e_3, g_1, h_1) and (d_3, f_5, g_4, h_2)

(63) (a_5, e_3, g_1, h_1) and (d_5, f_6, g_4, h_2)

(64) (a_5, e_3, g_1, h_1) and (b_5, e_6, g_2, h_1)

(65) (d_3, f_5, g_4, h_2) and (d_5, f_6, g_4, h_2)

(66) (d_3, f_5, g_4, h_2) and (b_5, e_6, g_2, h_1)

(67) (d_5, f_6, g_4, h_2) and (b_5, e_6, g_2, h_1)

(68) (a_5, e_3, g_1) and (b_5, e_6, g_2)

(69) (d_3, f_5, g_4) and (d_5, f_6, g_4)

(70) (d_3, f_5) and (d_5, f_6)

The minimal sets of parallel paths can be obtained by using the minimal covering table as shown in Table 6.5.3. The above seventy sets of parallel paths correspond to the columns of Table A.2.1. While its rows correspond to the letters that are shown at the inputs of the gates of Figure 6.5.4. Some of the minimal sets that can be obtained from Table A.2.1 are:

$(f_2, f_5, g_1, g_3, h_2)$, $(f_2, f_6, g_1, g_3, h_2)$, $(f_2, f_5, g_2, g_3, h_2)$,

$(f_2, f_6, g_2, g_3, h_2)$, $(f_1, f_5, g_1, g_3, h_2)$, $(f_1, f_6, g_1, g_3, h_2)$,

$(f_1, f_5, g_2, g_3, h_2)$, $(f_1, f_6, g_2, g_3, h_2)$, $(d_3, f_2, g_1, g_5, h_2)$,

$(d_5, f_2, g_1, g_3, h_2)$, $(d_3, f_2, g_2, g_3, h_2)$, $(d_5, f_2, g_2, g_3, h_2)$,

$(d_3, f_1, g_1, g_3, h_2)$, $(d_5, f_1, g_1, g_3, h_2)$, $(d_3, f_1, g_2, g_3, h_2)$,

$(d_5, f_1, g_2, g_3, h_2)$, $(d_3, f_1, g_1, g_3, h_1)$, $(d_5, f_1, g_1, g_3, h_1)$,

$(f_1, f_5, g_1, g_3, h_1)$, $(f_1, f_6, g_1, g_3, h_1)$, $(d_3, f_2, g_1, g_3, h_1)$,

$(d_5, f_2, g_1, g_3, h_1)$, $(f_2, f_5, g_1, g_3, h_1)$, $(f_2, f_6, g_1, g_3, h_1)$,

$(d_3, f_2, g_2, g_3, h_1)$, $(d_5, f_2, g_2, g_3, h_1)$, $(f_2, f_5, g_2, g_3, h_1)$,

$(f_2, f_6, g_2, g_3, h_1)$, $(d_3, f_1, g_2, g_3, h_1)$, $(d_5, f_1, g_2, g_3, h_1)$,

$(f_1, f_5, g_2, g_3, h_1)$ and $(f_1, f_6, g_2, g_3, h_1)$.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70
		x	x	x		x	x	x	x				x																																																								
																x	x		x	x	x	x	x				x																																										
																														x	x		x	x	x	x	x				x																												
																																											x		x	x	x	x			x	x			x														
																																																								x	x		x	x	x	x	x				x		
x																																																																						
		x	x		x	x			x		x	x	x																																																								
																x	x	x		x			x		x	x	x																																										
																														x	x	x		x			x		x	x	x																												
																																											x	x	x		x			x		x	x		x														
																																																								x	x	x		x			x		x	x	x		
	x																																																																					
		x	x	x	x		x			x	x			x	x																																																						
																x		x	x	x	x			x	x			x	x																																								
		x		x	x	x		x		x		x		x	x																																																					
																x	x	x	x			x		x		x		x	x																																								
																														x		x	x	x	x			x	x			x																											
																																											x	x		x	x	x	x	x				x															
																														x	x		x			x		x		x		x																											
																																											x	x	x	x			x		x		x	x															
																																																								x		x	x	x	x			x	x			x	x
	x																																																																					
																																																								x	x	x	x			x		x		x		x	x
x																																																																						
		x	x	x		x	x	x	x				x			x	x		x	x	x	x	x				x																																										
																														x	x		x	x	x	x	x				x		x		x	x	x	x			x	x			x														
x																																																								x	x		x	x	x	x	x				x		
		x	x		x	x			x		x	x	x			x	x	x		x			x		x	x	x																																										
																														x	x	x		x			x		x	x	x		x	x	x		x					x	x		x														
	x																																																							x		x		x		x	x		x	x	x		
		x	x	x	x		x			x	x			x	x	x		x	x	x	x			x	x			x	x																																								
		x		x	x	x		x		x		x		x	x	x	x	x	x			x		x		x		x	x																																								
																														x		x	x	x	x			x	x			x	x	x		x	x	x	x	x				x															
																														x	x	x	x			x		x		x		x	x	x		x			x		x		x	x															
	x																																																							x		x	x		x			x	x			x	x
x																																																								x	x	x	x	x		x		x		x		x	x
x		x		x		x	x	x	x				x			x	x	x	x	x	x	x	x				x			x	x		x	x	x	x	x				x		x		x	x	x	x			x	x			x	x	x		x	x	x	x	x				x		
	x	x	x		x	x					x	x	x			x	x			x	x		x		x	x	x			x	x	x		x			x		x	x	x		x	x	x		x			x		x	x		x	x	x	x		x			x		x	x	x		
		x	x	x	x	x	x	x	x		x	x		x		x	x	x	x	x		x		x	x	x		x		x		x	x	x	x			x	x			x	x	x		x	x	x	x	x				x															
x	x																													x	x	x	x			x		x		x		x	x	x	x	x			x		x		x	x		x	x	x	x	x	x	x		x	x	x		x	
x	x	x	x	x	x	x	x	x	x		x	x				x	x	x	x	x	x	x	x		x	x				x	x	x	x	x	x	x	x		x	x			x	x	x	x	x	x		x	x		x			x	x	x	x	x	x	x	x		x	x			
x	x	x	x	x	x	x	x	x	x	x	x	x				x	x	x	x	x	x	x		x	x	x				x	x	x	x	x	x	x		x	x	x			x	x	x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x		x	x	x			

TABLE A.2.1

Minimal Covering Table for Example 6.5.3

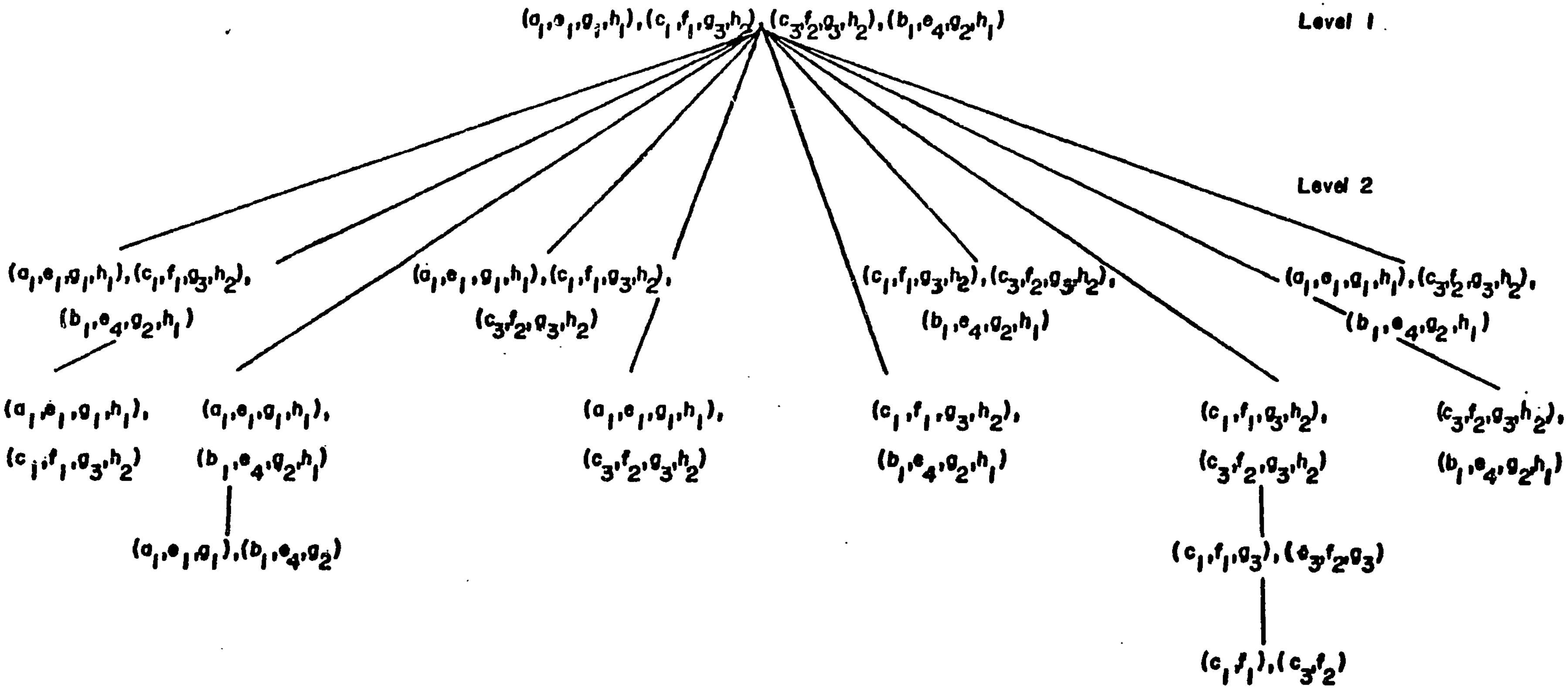


FIGURE A.2.1: Searching Tree for Set $[(a_1,e_1,g_1,h_1),(c_1,f_1,g_3,h_2),(c_3,f_2,g_3,h_2),(b_1,e_4,g_2,h_1)]$.

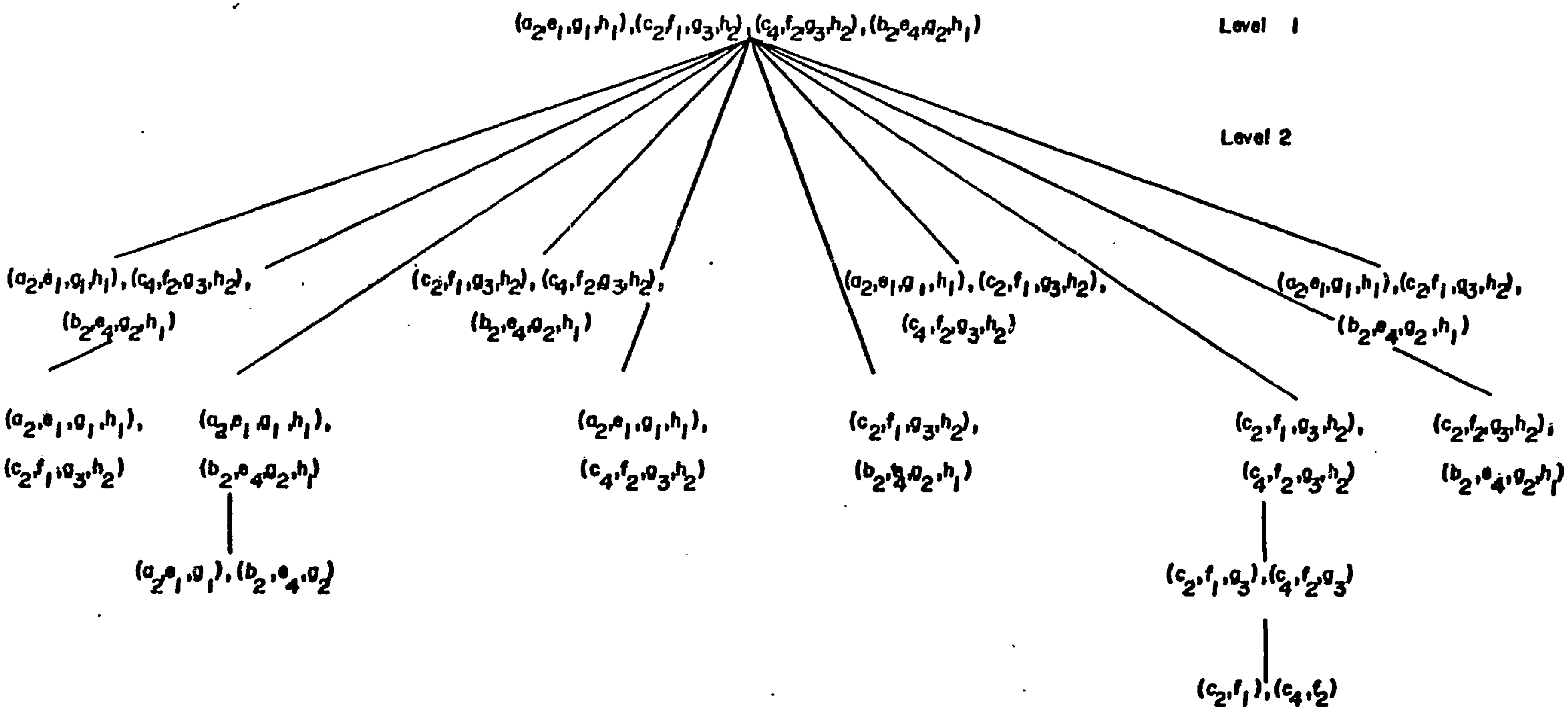


FIGURE A.2.2: Searching Tree for Set $[(a_2,e_1,g_1,h_1),(c_2,f_1,g_3,h_2),(e_4,f_2,g_3,h_2),(b_2,e_4,g_2,h_1)]$.

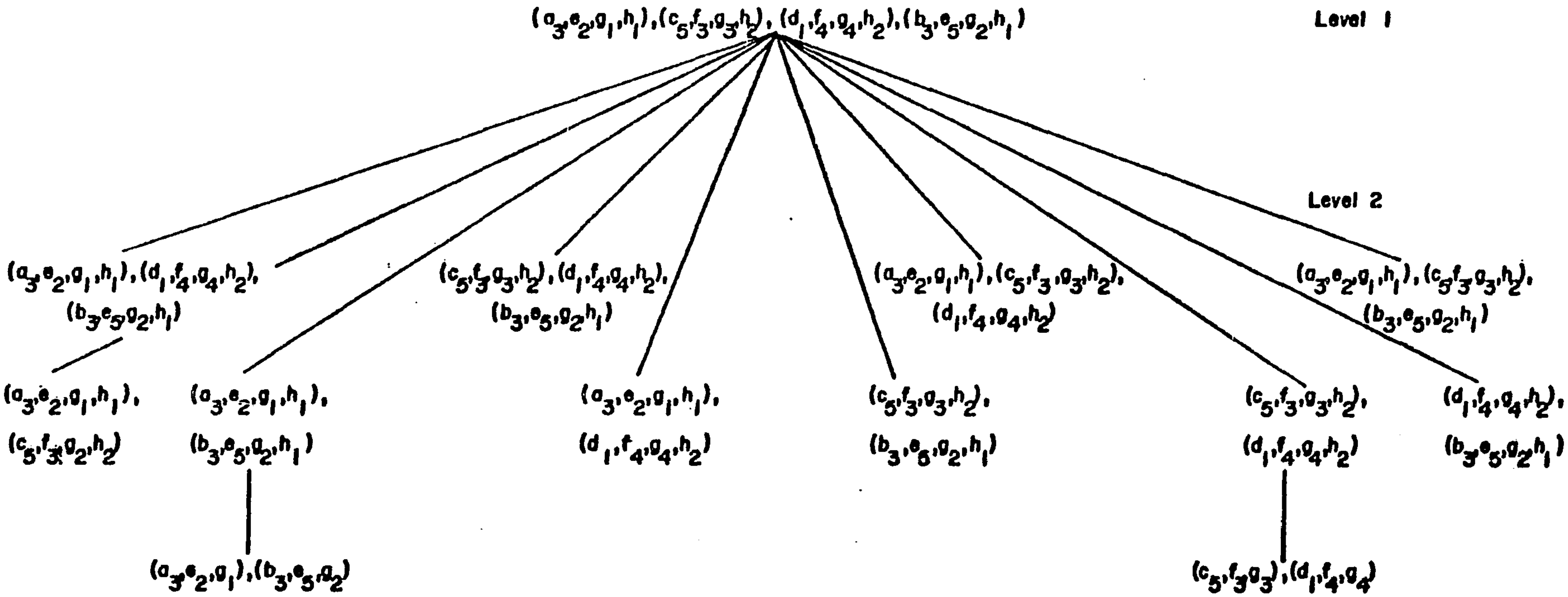


FIGURE A.2.3: Searching Tree for Set $[(a_3,e_2,g_1,h_1),(c_5,f_3,g_3,h_2),(d_1,f_4,g_4,h_2)(b_3,e_5,g_2,h_1)]$.

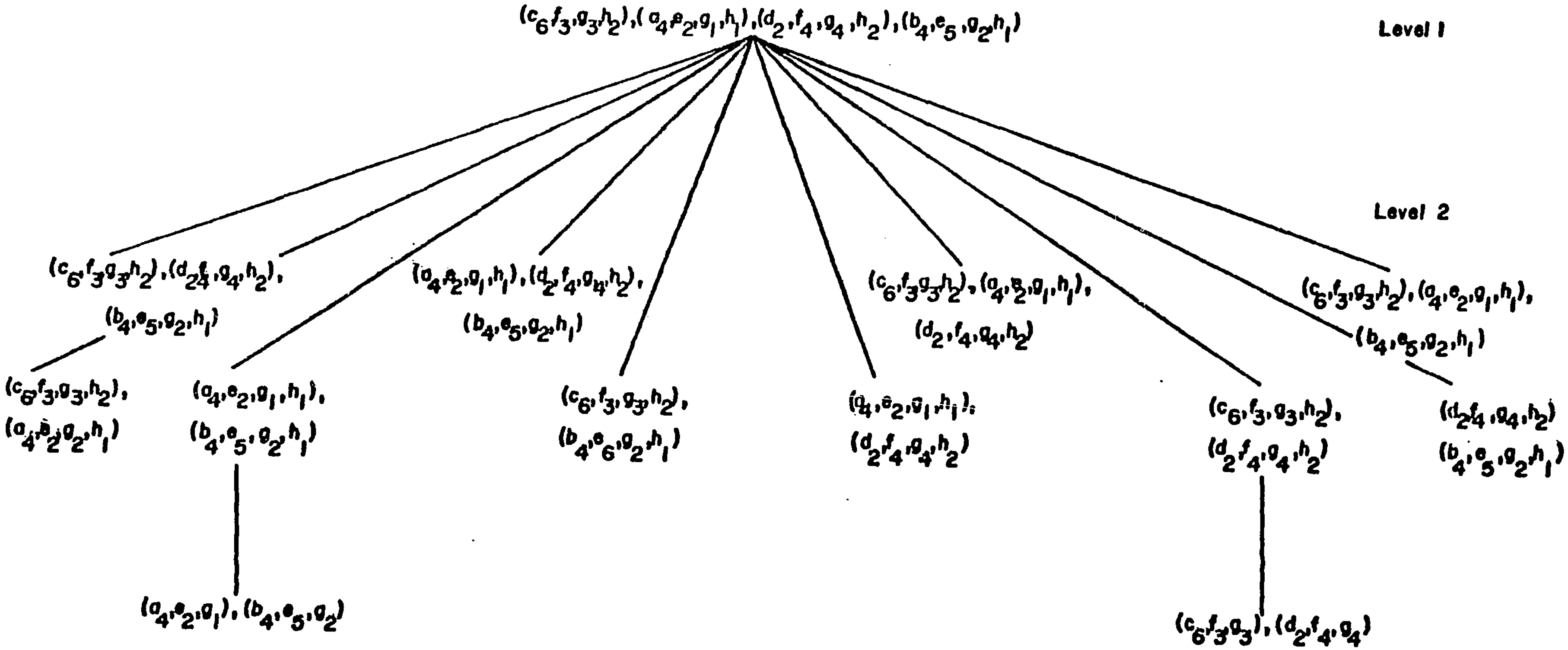


FIGURE A.2.4: Searching Tree for Set $[(c_6,f_3,g_3,h_2),(a_4,e_2,g_1,h_1),(d_2,f_4,g_4,h_2)(b_4,e_5,g_2,h_1)]$.

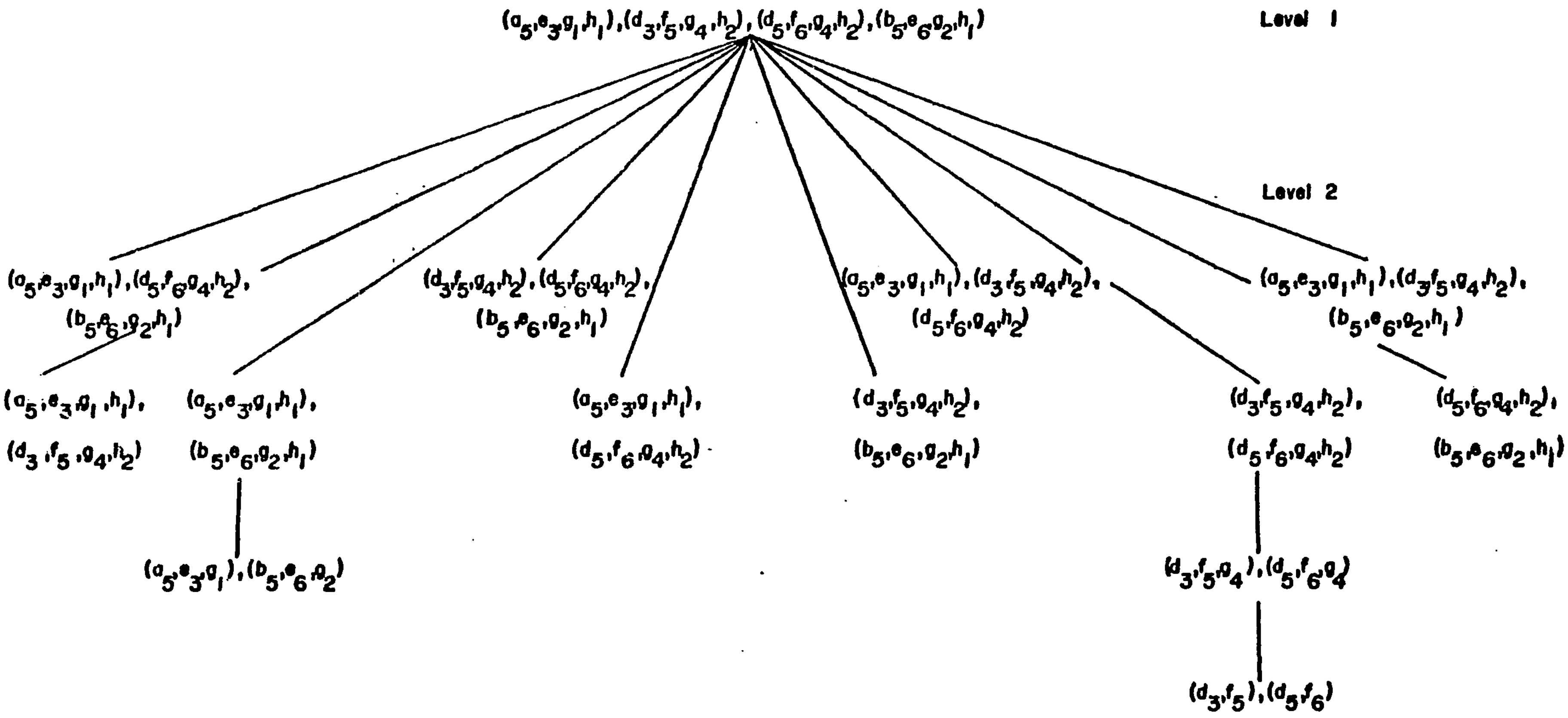


FIGURE A.2.5: Search Tree for Set $[(a_5,e_3,g_1,h_1),(d_3,f_5,g_4,h_2),(d_5,f_6,g_4,h_2),(b_5,e_6,g_2,h_1)]$.

REFERENCES

In the following references, STAMLD stands for the Symposia on the theory and Applications of Multiple-Valued Logic Design and ISML for the International Symposia on Multiple-Valued Logic.

1. Reed, I.S.: A Class of Multiple-Error-Correcting Codes and the Decoding Scheme. IRE Trans., IT-4, 1954, pp. 38-49.

2. Akers, S.B., Jr.: On a Theory of Boolean Functions, J. Soc. Indust. Appl. Math., Vol. 7, No. 4, Dec. 1959, pp. 487-498.

3. Sellers, F.F.; Hsiao, M.Y.; and Bearnson, L.W.: Analyzing Errors with the Boolean Difference. IEEE Trans. on Computer, Vol. C-17, July 1968, pp. 676-683.

4. Thayse, A.: Boolean Differential Calculus. Phillips Res. Repts., 26, 1971, pp. 229-246.

5. Lee, S.C.; and Keren-Zvi, Y.: A Generalized Boolean Algebra and its Applications to Logic Design. ISML, 1975, pp. 88-98.

6. Lee, S.C.: Vector Boolean Algebra and Calculus, IEEE Trans. on Computer, Vol. C-25, September 1976, pp. 865-874.

7. Lee, S.C.: Modern Switching Theory and Digital Design. Prentice-Hall, 1978.

8. Post, E.L." Introduction to a General Theory of Elementary Propositions. Amer. J. Math., Vol. 43, 1921, pp. 163-185.

9. Lukasiewicz, J.: O. Logic Trojwartoscioweg. Buch Filozoficzny, Vol. 5, 1920, pp. 169-171.

10. Rosenbloom, P.C.: Post Algebra I. Postulates and General Theory. Amer. J. Math., Vol. 64, 1942, pp. 167-188.

11. Vranesic, Z.; Lee, E.; and Smith, K.C.: A Many Valued Algebra for Switching Systems. IEEE Trans. on Computer, October 1970, pp. 964-971.

12. Su, S.Y.; and Sarris, A.A.: The Relationship Between Multi-Valued Algebra and Boolean Algebra Under Different Definitions of Complement. IEEE Trans. on Computer, Vol. C-21, May 1972, pp. 479-485.

13. Lee, S.C.; Ajabnoor, Y.M.: Digital Calculus, ISML, 1978, pp. 128-141.

14. Resher, N.: Many-Valued Logic. McGraw-Hill, New York, 1969.

15. Vranesic, Z.; Smith, K.C.: Engineering Aspects of Multi-Valued Logic Systems. Computer J., Vol. 7, September 1974, pp. 34-41.

16. Abraham, G.: Variable Radius Multi-Valued Integrated Circuits. Computer J., Vol. 7, September 1974, pp. 42-59.

17. Allen, G.M.; and Givone, D.D.: A Minimization Technique for Multiple-Valued Logic Systems. IEEE Trans. on Computer, Vol. C-21, February 1968, pp. 182-184.

18. Lee, S.C., Digital Circuits and Logic Design. Prentice-Hall, 1976.

19. Boole, G.: An Investigation of the Laws of Thought. New York: Dover, 1854, 1940, 1958.

20. Sheng, C.L.: Introduction to Switching Logic. International Textbook Company, 1972.

21. Quine, W.V.: The Problem of Simplifying Truth Functions. Am. Math. Monthly, 59, No. 8, October 1952, pp. 521-531.

22. McCluskey, E.J.: Minimization of Boolean Functions. Bell System Tech. J., 35, No. 5, November 1956, pp. 1417-1444.

23. Marions, P.N.: Derivation of Minimal Complete Sets of Test Input Sequences Using Boolean Differences. IEEE Trans. On Computer, January 1971, Vol. C-20.

24. Marions, P.; Page, E.; Thomason, M.: Fault Detection in Sequantial Networks Using Time Dependent Boolean Differences. 1971 Proc. IEEE Int. Computer Soc. Conf., pp. 79-80.

25. Yau, S.S.; Tang, Y.S.: An Efficient Algorithm for Generating Complete Sets for Combinational Logic Circuits. IEEE Trans. on Computer, Vol. C-20, November 1971, pp. 1245-1251.

26. Thayse, A.: A Variational Diagnosis Method for Stuck-Faults in Combinational Networks. Philips Res. Repts., 27, 1972, pp. 82-98.

27. Sheppard, D.A.: An Application of Many Valued Logics to Fault Detection, 1973 ISML, pp. 192-204.

28. Rescher, N.: Many-Valued Logic. McGraw-Hill, N.Y., 1969.

29. Rine, D.C., (ed.): Computer Science and Multi-Valued Logic. North-Holland, 1977.

30. Brozowski, J.A.; and Yoeli, M.: Digital Networks, Prentice-Hall, 1976.

31. Vranesic, Z.; and Waliuzzaman, K.M.: Functional Transformation in Simplification of Multi-Valued Switching Functions. IEEE Trans. on Computer, January 1972, pp. 102-105.

32. Huffman, D.: The Design and Use of Hazard-Free Switching Networks. J. ACM 4, 1957, pp. 47-62.

33. Eichelberger, E.B.: Hazard Detection in Combinational and Sequantial Switching Circuits. IBM J., 1965, pp. 90-99.

34. Bredeson, J.G.; and Hulina, P.T.: Elimination of Static and Dynamic Hazards for Multiple-Input Changes in Combinational Switching Circuits. Inf. and Cont., Vol. 20, No. 2, March 1972, pp. 114-124.

35. Mukaidono, M.: The B-ternary Logic and its Applications to the Detection of Hazards in Combinational Switching Circuits. ISML, 1978, pp. 269-275.

36. Mete, L.L.; Das, S.; and Chung, Y.H.: A Logic Hazard Detection and Elimination Method. Inf. and Cont., Vol. 26, 1974, pp. 351-368.

37. Thayse, A.: Transient Analysis of Logical Networks Applied to Hazard Detection. Phillips Res. Repts., Vol. 25, 1970, pp. 261-368.

38. Yoeli, M.; and Rinon, S.: Application of Ternary Algebra to the Study of Static Hazards. ACM J., 11, No. 7, 1969, pp. 84-97.

39. Beister, J.: A Unified Approach to Combinational Hazards. IEEE Trans. on Computer, Vol. C-23, No. 6, 1974, pp. 566-575.

40. Armstrong, D.B.: On Finding a Minimal Set of Fault Detection Tests for Combinational Logic Nets. IEEE Trans. Electronic Computer, Vol. CE-15, February 1966, pp. 66-75.

41. Lee, S.C.; and Lee, E.T.: Multi-Valued Symmetric Function. STAMD, 1971, pp. 62-83.

42. Allen, C.M.; and Givone, D.D.: A Minimization Technique for Multiple-Valued Logic Systems. IEEE Trans. on Computer, Vol. C-17, February 1968, pp. 182-184.