ALGEBRAIC MINIMIZATION AND THE DESIGN OF TWO-TERMINAL CONTACT NETWORKS

by

EDWARD JOSEPH McCLUSKEY, JR.

A.B., Bowdoin College
(1953)

B.S., M.S., Massachusetts Institute of Technology
(1953)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF
SCIENCE

at the

MASSACHUSETTS INSTITUTE OF
TECHNOLOGY

June, 1956

Signature of Author
Department of Electrical Engineering,
April 30, 1956

Certified by
Thesis Supervisor

Accepted by
Chairman, Departmental Committee
on Graduate Students

DEPARTMENT OF ELECTRICAL ENGINEERING

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE 39, MASSACHUSETTS

May 4, 1956

Mr. E. J. McCluskey
Bell Telephone Laboratories
Whippany, New Jersey

Dear Mr. McCluskey:

This letter is to give you permission to print additional copies of your thesis by the multilith process, and to submit to the Department of Electrical Engineering copies thus produced, in lieu of the typed copies normally required.

A copy of this letter is to be reproduced by the same process and is to be placed in each copy of the thesis immediately following its title page.

Sincerely yours,

S. H. Caldwell
for the
Department Graduate Committee

SHC:mm

Algebraic Minimization and the Design of Two-Terminal Contact Networks

by

Edward Joseph McCluskey, Jr.

Submitted to the Department of Electrical Engineering on April 30, 1956 in partial fulfillment of the requirements for the degree of Doctor of Science.

Systems such as digital computers, telephone central offices, etc. are commonly constructed so that they operate with signals which only take on two values. The circuits used in building such systems are called switching circuits and are described mathematically by means of a two-valued algebra. In designing switching circuits the circuit requirements are reduced to algebraic expressions and then these expressions are simplified by means of the theorems of Boolean Algebra. The aim of algebraic simplification is to reduce the amount of equipment needed in the circuit.

The first part of this thesis presents a systematic procedure for writing a Boolean function in the simplest sum of products form. This form corresponds directly to the two-stage, single-output diode logic circuit using the fewest possible diodes. The procedure developed can be programmed on a digital computer in order to handle functions of large numbers of variables.

In carrying out the procedure of part one it is helpful to know if the function being simplified remains unchanged when some of its variables are permuted (or complimented). A method for the detection of such invariance (called group invariance) has therefore been developed and is presented in Appendix A. An extension of this method for determining if a function is totally symmetric (invariant under any permutation of its variables) is also presented. It is important to know whether a function possesses invariance properties since special design methods may then be applicable.

In part two of the thesis a method is developed for designing a two-terminal contact network. For relay switching circuits, contact networks must be designed to control the operation of the circuit relays and to provide output signals. Existing methods of contact network design are all unsatisfactory either because they are unsystematic or because they restrict the form of the network being designed. The method developed here uses the algebraic expression derived by the procedure of part one as a starting point and consists of directly providing a path through the network for each term of the algebraic expression. Methods are presented for determining how to form each path economically.

Thesis Supervisor: S. H. Caldwell
Title: Professor of Electrical Engineering

ACKNOWLEDGEMENTS

The author wishes to acknowledge his indebtedness to Professor S. H. Caldwell who supervised the preparation of this thesis and from whom the author learned the basic principles of switching theory. The research reported here was influenced strongly by many discussions with Professor D. A. Huffman who contributed much to the author's understanding of the problems to be considered. In addition, Messrs. S. H. Unger, W. J. Cadden, and G. H. Mealy contributed helpful comments.

Special mention should be made of Professor W. K. Linvill without whose advice and encouragement this research would probably never have been undertaken.

This research was supported in part by the Signal Corps; the Office of Scientific Research, Air Research and Development Command; the Office of Naval Research; and the Bell Telephone Laboratories.

The author wishes to express his thanks to Miss Arlene McKay who did the final typing.

Last, but far from least, the author must express gratitude to his wife, Roberta, for her assistance in the actual writing and editing of the thesis as well as for her sustaining faith throughout the entire period of research.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

INTRODUCTION

Digital computer circuits, automatic telephone exchange circuits, and automatic traffic control circuits are examples of switching circuits. The distinguishing feature of these circuits is that they operate with binary signals; that is, signals which have only two stable values (off or on, high or low, ground or no ground, etc.). Such circuits can be constructed with relays, diodes, vacuum tubes, transistors, magnetic amplifiers, etc. The theory to be developed here deals with general properties of switching circuits which are independent of their particular physical components.

A combinational circuit is one in which the outputs at a given instant of time are determined only by the inputs at the same instant of time; when the outputs depend also on previous inputs, the circuit is called sequential. The problem of reducing the specifications for a sequential circuit to specifications for several combinational circuits has effectively been solved [H1].

In 1938, Shannon [S1] showed that Boolean Algebra could be used to design combinational switching circuits (contact networks). In general form, the design procedure consists of: (1) writing the contact-network requirements as an algebraic expression, (2) simplifying this expression by use of the theorems of Boolean Algebra, (3) determining a contact network whose transmission equals the simplified algebraic expression.

A systematic method for writing an algebraic expression which corresponds to a given set of circuit requirements was presented by Montgomerie [MN1]. He introduced the idea of a table of combinations (called a truth table by logicians).

Methods for simplifying algebraic expressions were published by Aiken [SCL1], Veitch [V1], Quine [Q1], and Karnaugh [K1]. These methods are similar in that they all result in the same type of expression (the simplest sum of products form) and they all fail when the expression to be simplified becomes sufficiently complex. Karnaugh's method is a chart method which is a refinement of the methods of Veitch and Aiken. It is very effective for functions involving few variables but becomes very difficult and unsystematic as the number of variables becomes large. Quine's method is an algebraic procedure which is rather unwieldy but which is sometimes useful for functions of many variables. The method to be presented in Part 1 of this thesis has certain similarities with both the chart and algebraic methods. However, it is usually simpler than either method for functions of more than six variables and can always be systematically carried to completion. In addition it appears that the method to be presented here can be programmed on a digital computer without requiring excessively large storage capacity or computing time.

In connection with this method for simplifying algebraic expression, a method for determining group invariance [S2] or total symmetry [C2] has been determined. This method is presented in Appendix A.

In Shannon's original paper [S1], the design procedure actually consisted of writing the simplest algebraic expression (in the sense that there are the fewest possible variable appearances) corresponding to the circuit requirements and then drawing the corresponding contact network directly. This procedure has two defects. There is no systematic method for obtaining the simplest algebraic expression and only series-parallel networks will result. Numerous examples exist of nonplanar or bridge-type networks which use significantly fewer contacts than the corresponding series-parallel network. In a second paper [S2], Shannon presented another method for designing a contact network. The success of this method, called the Shannon MN synthesis, depends on the designer's ability to recognize permissible circuit modifications which very frequently destroy the series-parallel character of the network while at the same time reducing the number of contacts included. While this method can result in nonplanar or bridge-type networks, it is not truly general since the structure of the resulting network is arbitrarily assumed to be of a certain general form. A matrix method of design has also been presented [HS1]. This method does not place any restrictions on the form of the final network, but it lacks any systematic procedure and in addition is cumbersome to actually carry out. The method for designing contact networks which is presented in Part 2 of this thesis is essentially different from any of the previous methods. The required algebraic expression is that obtained by the method of Part 1. Since no restriction is placed on the form of the final network, this is a general method. A systematic approach is given and all algebraic manipulations are correlated with the corresponding effects on the actual network.

PART 1

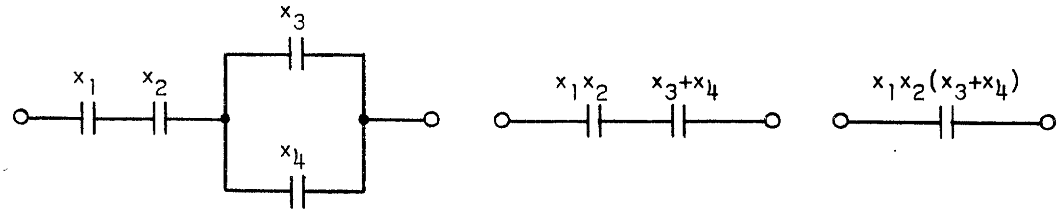
ALGEBRAIC MINIMIZATION

1.1 Algebraic Symbolism For Relay Contact Networks

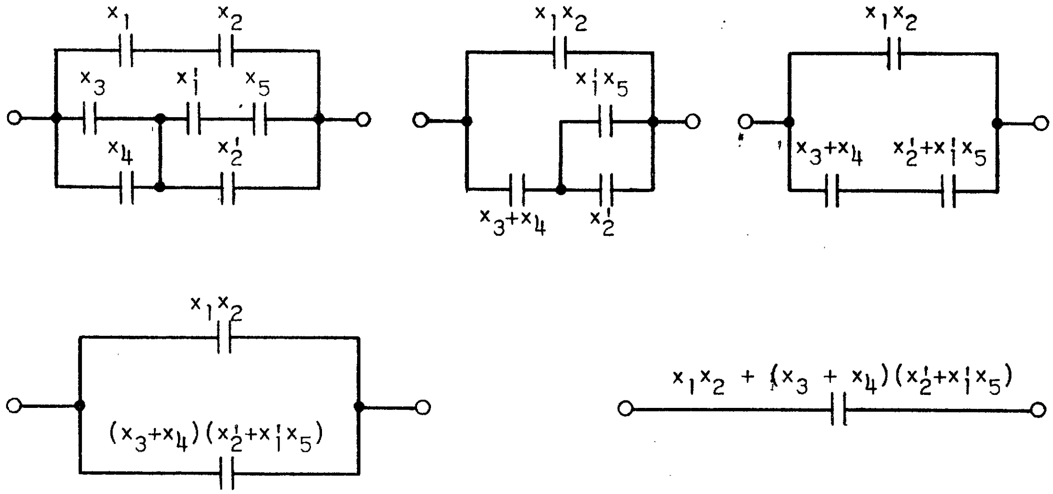
Each relay will be denoted by a symbol such as X_1, Y_2, etc. Contacts belonging to the X_1 relay will be designated by x_1 if normally open and by x_1' if normally closed. It is common practice to interpret these contact designators as transmissions by setting $x_1 = 0$, $x_1' = 1$ when relay X_1 is unoperated and $x_1 = 1$, $x_1' = 0$ when relay X_1 is operated. The symbols 0 and 1 represent open and closed circuits respectively.

A series connection of contacts x_1 and x_2 has the transmission x_1x_2: This function equals one only when both x_1 and x_2 equal one, and the series connection of x_1 and x_2 is a closed circuit only when both x_1 and x_2 are closed circuits. Similarly, the transmission of a parallel connection of contacts x_1 and x_2 can be written as $x_1 + x_2$, if it is postulated that $1+1=1$. This postulate is a statement of the fact that two closed circuits in parallel are indistinguishable from a single closed circuit.

The over-all transmission of any planar, nonbridge (series-parallel) network can be written directly. See Fig. 1.1-1. First, transmissions are written for all series or parallel connections of contacts, then each such connection is replaced by a single contact designated by the appropriate transmission. This process is repeated until only one contact remains. The associated designator is the desired over-all transmission.



(a) $T = x_1x_2(x_3 + x_4)$



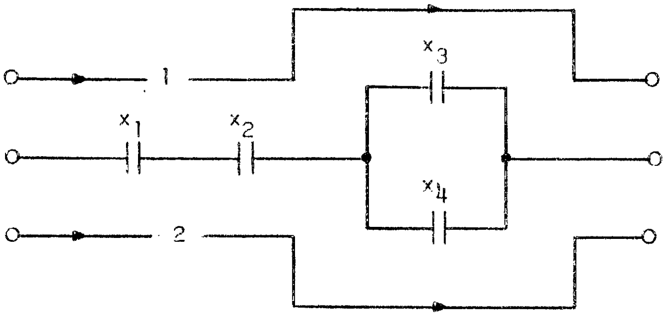
(b) $T = x_1x_2 + (x_3 + x_4)\ (x_2' + x_1'x_5)$

Figure 1.1-1

Determination of Overall Transmission for Series-Parallel Networks.

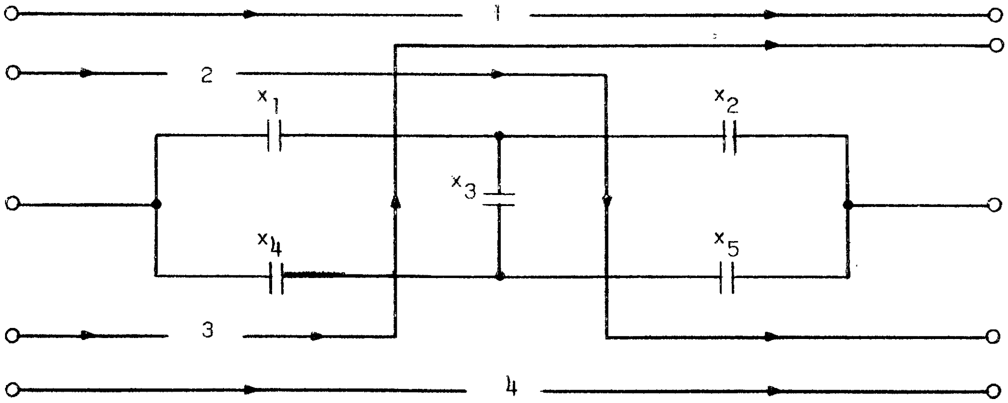
This method is not directly applicable to nonseries-parallel networks. Another method, which works for all networks, consists of tracing all paths from input to output, writing the transmission of each path, and then forming the over-all transmission as the sum of the individual path transmissions. See Fig. 1.1-2. Each path transmission will be a product of literals, where a literal is defined as a variable with or without the associated prime (x_1, x_2' are literals). The input and output terminals are connected together only when all contacts of at least one of the paths through the network are closed. The over-all transmission can equal one only when at least one of the product terms equals one; that is, when all literals of at least one product terms are equal to one.

It has been specifically postulated that $1+1=1$. Other postulates which have been implicitly assumed are: $0+1=1+0=1$, $0+0=0$, $0 \cdot 0=0$, $0 \cdot 1=1 \cdot 0=0$, $1 \cdot 1=1$, $0'=1$, $1'=0$, and $x_1 = 0$ if $x_1 \neq 1$, $x_1 = 1$ if $x_1 \neq 0$. Since these postulates are identical with the postulates of Boolean algebra, the switching algebra developed here is actually a Boolean Algebra and all the Boolean Algebra theorems apply. This was pointed out by Shannon in a paper which discusses the theorems in detail [S1]. Table 1.1-1 is a list of the Boolean Algebra theorems commonly used.



$$T = \underset{(1)}{x_1x_2x_3} + \underset{(2)}{x_1x_2x_4}$$

(a) Series-Parallel Network



$$T = \underset{(1)}{x_1x_2} + \underset{(2)}{x_1x_3x_5} + \underset{(3)}{x_4x_3x_2} + \underset{(4)}{x_4x_5}$$

(b) Bridge Network

Figure 1.1-2

Determination of Overall Transmission by Path Tracing.

TABLE 1.1-1 Boolean Algebra Theorems

(1) $0 + x_1 = x_1$

(1') $1 \cdot x_1 = x_1$

(2) $1 + x_1 = 1$

(2') $0 \cdot x_1 = 0$

(3) $x_1 + x_1 = x_1$

(3') $x_1 \cdot x_1 = x_1$

(4) $(x_1)' = x_1'$

(4') $(x_1')' = x_1$

(5) $x_1 + x_1' = 1$

(5') $x_1 \, x_1' = 0$

(6) $x_1 + x_2 = x_2 + x_1$

(6') $x_1 \, x_2 = x_2 \, x_1$

(7) $x_1 + x_1 x_2 = x_1$

(7') $x_1 \, (x_1 + x_2) = x_1$

(8) $(x_1 + x_2') \, x_2 = x_1 \, x_2$

(8') $x_1 \, x_2' + x_2 = x_1 + x_2$

(9) $x_1 \, x_2' + x_1 \, x_2 = x_1$

(9') $(x_1 + x_2) \, (x_1 + x_2') = x_1$

(10) $(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3)$

(10') $x_1 \, (x_2 \, x_3) = (x_1 \, x_2) \, x_3$

(11) $x_1 \, x_2 + x_1 \, x_3 = x_1 \, (x_2 + x_3)$

(11') $(x_1 + x_2) \, (x_1 + x_3) = x_1 + x_2 \, x_3$

(12) $(x_1 + x_2) \, (x_1' + x_3) \, (x_2 + x_3) = (x_1 + x_2) \, (x_1' + x_3)$

(12') $x_1 \, x_2 + x_1' \, x_3 + x_2 \, x_3 = x_1 \, x_2 + x_1' \, x_3$

(13) $(x_1 + x_2) \, (x_1' + x_3) = x_1 \, x_3 + x_1' \, x_2$

(14) $(x_1 + x_2 + \cdots + x_n)' = x_1' \, x_2' \cdots x_n'$

(14') $(x_1 \, x_2 \cdots x_n)' = x_1' + x_2' + \cdots + x_n'$

(15) $f(x_1, x_2, \cdots, x_n) = x_1 \cdot f(1, x_2, \cdots, x_n) + x_1' \cdot f(0, x_2, \cdots, x_n)$

(15') $f(x_1, x_2, \cdots, x_n) = [x_1 + f(0, x_2, \cdots, x_n] \cdot [x_1' + f(1, x_2, \cdots, x_n)]$

(16) $f(x_1, x_2, \cdots, x_n, +, \cdot)' = f(x_1', x_2', \cdots, x_n', \cdot, +)$

	x_1	x_2	x_3	T
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

(a)

Table of Combinations

$x_1' \; x_2' \; x_3'$

$x_1' \; x_2' \; x_3$

$x_1' \; x_2 \; x_3'$

$x_1' \; x_2 \; x_3$

$x_1 \; x_2' \; x_3'$

$x_1 \; x_2' \; x_3$

$x_1 \; x_2 \; x_3'$

$x_1 \; x_2 \; x_3$

(b)

p-terms

$$T = x_1'x_2'x_3 + x_1'x_2x_3' + x_1'x_2x_3 + x_1x_2'x_3' + x_1x_2'x_3 + x_1x_2x_3'$$

(c)

Standard Sum

Figure 1.1-3 Circuit Specifications

For each row of the table of combinations a transmission can be written which equals "one" only when the variables have the values listed in that row of the table. These transmissions will be called elementary product terms (or more simply, p-terms) since any transmission can always be written as a sum of these p-terms. Figure 1.1-3b lists the p-terms for Fig. 1.1-3a. Note that every variable appears in each p-term.

The p-term corresponding to a given row of a table of combinations is formed by priming any variables which have a "zero" entry in that row of the table and by leaving unprimed those variables which have "one" entries. It is possible to write an algebraic expression for the over-all circuit transmission directly from the table of combinations. This over-all transmission, T, is the sum of the p-terms corresponding to those rows of the table of combinations for which T is to have the value "one." See Fig. 1.1-3c. Any transmission which is a sum of p-terms is called a standard sum [C1].

The decimal numbers in the first column of Fig. 1.1-3a are the decimal equivalents of the binary numbers formed by the entries of the table of combinations. A concise method for specifying a transmission function is to list the decimal numbers of those rows of the table of combinations for which the function is to have the value one. Thus the function of Fig. 1.1-3 can be specified as $\sum(1,2,3,4,5,6)$.

1.2 The Minimum Sum

By use of the theorem $x_1x_2 + x_1' x_2 = x_2$ it is possible to obtain from the standard sum other equivalent sum functions; that is, other sum functions which correspond to the same table of combinations. These functions are still sums of products of literals but not all of the variables appear in each term. For example, the transmission of Fig. 1.1-3, $T = x_1'x_2'x_3 + x_1'x_2x_3' + x_1'x_2x_3 + x_1x_2'x_3' + x_1x_2'x_3 + x_1x_2x_3' =$ $(x_1'x_2'x_3 + x_1'x_2x_3) + (x_1'x_2x_3' + x_1x_2x_3') + (x_1x_2'x_3' + x_1x_2'x_3) =$ $(x_1'x_2'x_3 + x_1x_2'x_3) + (x_1'x_2x_3' + x_1'x_2x_3) + (x_1x_2'x_3' + x_1x_2x_3')$ can be written as either $T = x_1'x_3 + x_2x_3' + x_1x_2'$ or $T =$ $x_2'x_3 + x_1'x_2 + x_1x_3'$.

The sum functions which have the fewest terms of all equivalent sum functions will be called <u>minimum</u> <u>sums</u> unless these functions having fewest terms do not all involve the same number of literals. In such cases, only those functions which involve the fewest literals will be called minimum sums. For example, the function $T = \sum(0,1,3,4,6,7,9,11,13,27)$ can be written as either $T= x_5'x_4'x_3'x_2' + x_5'x_4'x_3'x_1' + x_5'x_4'x_2'x_1 +$ $x_5'x_4x_2'x_1 + x_4x_3'x_2x_1$ or as $T = x_5'x_4'x_2'x_1' + x_5'x_4'x_3x_2 + x_5'x_4x_2'x_1 +$ $x_4x_3'x_2x_1 + x_5'x_3'x_1$. Only the second expression is a minimum sum since it involves 19 literals while the first expression involves 20 literals.

In principle it is possible to obtain a minimum sum for any given transmission by enumerating all possible equivalent sum functions then selecting those functions which have the fewest terms, and finally selecting from these the functions which contain fewest literals. Since the number of equivalent sum functions may be quite large, this procedure is not generally practical. The following sections present a practical method for obtaining a minimum sum without resorting to an enumeration of all equivalent sum functions.

1.3 Prime Implicants

When the theorem $x_1x_2 + x_1x_2' = x_1$ is used to replace by a single term, two p-terms, which correspond to rows i and j of a table of combinations, the resulting term will equal "one" when the variables have values corresponding to either row i or row j of the table. Similarly, when this theorem is used to replace, by a single term, a term which equals "one" for rows i and j and a term which equals "one" for rows k and m, the resulting term will equal "one" for rows i, j, k and m of the table of combinations. A method for obtaining a minimum sum by repeated application of this theorem ($x_1x_2' + x_1x_2 = x_1$) was first presented by Quine [Q1]. In this method, the theorem is applied to all possible pairs of p-terms, then to all possible pairs of the terms obtained from the p-terms, and so on, until no further applications of the theorem are possible. It may be necessary to pair one term with several other terms in applying this theorem. In Example 1.3-2 the theorem is applied to the terms labeled 5 and 7 and also to the terms labeled 5 and 13. All terms paired with other terms in applying the theorem are then discarded. The remaining terms are called prime implicants [Q1]. Finally a minimum sum is formed as the sum of the fewest prime implicants which when taken together will equal "one" for all required rows of the table of combinations. The terms in the minimum sum will be called minimum sum terms or ms-terms.

Example 1.3-1

$$T = \sum(3,7,8,9,12,13)$$

Standard Sum:

$$T = \underset{\begin{bmatrix}0\ 0\ 1\ 1\\ 3\end{bmatrix}}{x_1'x_2'x_3x_4} + \underset{\begin{bmatrix}0\ 1\ 1\ 1\\ 7\end{bmatrix}}{x_1'x_2x_3x_4} + \underset{\begin{bmatrix}1\ 0\ 0\ 0\\ 8\end{bmatrix}}{x_1x_2'x_3'x_4'} + \underset{\begin{bmatrix}1\ 0\ 0\ 1\\ 9\end{bmatrix}}{x_1x_2'x_3'x_4} + \underset{\begin{bmatrix}1\ 1\ 0\ 0\\ 12\end{bmatrix}}{x_1x_2x_3'x_4'} + \underset{\begin{bmatrix}1\ 1\ 0\ 1\\ 13\end{bmatrix}}{x_1x_2x_3'x_4}$$

Note: The bracketed binary and decimal numbers below the sum terms indicate the rows of the table of combinations for which the corresponding term will equal "one". A binary character in which a dash appears represents the two binary numbers which are formed by replacing the dash by a "0" and then by a "1". Similarly a binary character in which two dashes appear represents the four binary numbers formed by replacing the dashes by "0" and "1" entries, etc.

$$\underset{\begin{bmatrix}0\ 0\ 1\ 1\\ 3\end{bmatrix}}{x_1'x_2'x_3x_4} + \underset{\begin{bmatrix}0\ 1\ 1\ 1\\ 7\end{bmatrix}}{x_1'x_2x_3x_4} = \underset{\begin{bmatrix}0\ -\ 1\ 1\\ 3,7\end{bmatrix}}{x_1'\quad x_3x_4}$$

$$\underset{\begin{bmatrix}1\ 0\ 0\ 0\\ 8\end{bmatrix}}{x_1x_2'x_3'x_4'} + \underset{\begin{bmatrix}1\ 0\ 0\ 1\\ 9\end{bmatrix}}{x_1x_2'x_3'x_4} = \underset{\begin{bmatrix}1\ 0\ 0\ -\\ 8,9\end{bmatrix}}{x_1x_2'x_3'}$$

$$x_1x_2x_3'x_4' + x_1x_2x_3'x_4 = x_1x_2x_3'$$

$$\begin{bmatrix}1\ 1\ 0\ 0\\ 12\end{bmatrix}\quad\begin{bmatrix}1\ 1\ 0\ 1\\ 13\end{bmatrix}\quad\begin{bmatrix}1\ 1\ 0\ -\\ 12,13\end{bmatrix}$$

$$x_1x_2'x_3' + x_1x_2x_3' = x_1\quad x_3'$$

$$\begin{bmatrix}1\ 0\ 0\ -\\ 8,9\end{bmatrix}\quad\begin{bmatrix}1\ 1\ 0\ -\\ 12,13\end{bmatrix}\quad\begin{bmatrix}1\ -\ 0\ -\\ 8,9,12,13\end{bmatrix}$$

Prime Implicants: $x_1\quad x_3'$, $x_1'\quad x_3x_4$

$$\begin{bmatrix}1\ -\ 0\ -\\ 8,9,12,13\end{bmatrix}\quad\begin{bmatrix}0\ -\ 1\ 1\\ 3,7\end{bmatrix}$$

Minimum Sum: $T = x_1x_3' + x_1'x_3x_4$

Example 1.3-2

$$T = \sum(5,7,12,13)$$

Standard Sum: $T = x_1'x_2x_3'x_4 + x_1'x_2x_3x_4 + x_1x_2x_3'x_4' + x_1x_2x_3'x_4$

$$\begin{bmatrix}0\ 1\ 0\ 1\\ 5\end{bmatrix}\quad\begin{bmatrix}0\ 1\ 1\ 1\\ 7\end{bmatrix}\quad\begin{bmatrix}1\ 1\ 0\ 0\\ 12\end{bmatrix}\quad\begin{bmatrix}1\ 1\ 0\ 1\\ 13\end{bmatrix}$$

$$x_1'x_2x_3'x_4 + x_1'x_2x_3x_4 = x_1'x_2\quad x_4$$

$$\begin{bmatrix}0\ 1\ 0\ 1\\ 5\end{bmatrix}\quad\begin{bmatrix}0\ 1\ 1\ 1\\ 7\end{bmatrix}\quad\begin{bmatrix}0\ 1\ -\ 1\\ 5,7\end{bmatrix}$$

$$x_1'x_2x_3'x_4 + x_1x_2x_3'x_4 = x_2x_3'x_4$$

$$\begin{bmatrix} 0\ 1\ 0\ 1 \\ 5 \end{bmatrix} \begin{bmatrix} 1\ 1\ 0\ 1 \\ 13 \end{bmatrix} \begin{bmatrix} -\ 1\ 0\ 1 \\ 5,13 \end{bmatrix}$$

$$x_1x_2x_3'x_4' + x_1x_2x_3'x_4 = x_1x_2x_3'$$

$$\begin{bmatrix} 1\ 1\ 0\ 0 \\ 12 \end{bmatrix} \begin{bmatrix} 1\ 1\ 0\ 1 \\ 13 \end{bmatrix} \begin{bmatrix} 1\ 1\ 0\ - \\ 12,13 \end{bmatrix}$$

Prime Implicants: $x_1'x_2\ \ x_4$, $x_2x_3'x_4$, $x_1x_2x_3'$

$$\begin{bmatrix} 0\ 1\ -\ 1 \\ 5,7 \end{bmatrix} \begin{bmatrix} -\ 1\ 0\ 1 \\ 5,13 \end{bmatrix} \begin{bmatrix} 1\ 1\ 0\ - \\ 12,13 \end{bmatrix}$$

Minimum Sum: $T = x_1'x_2x_4 + x_1x_2x_3'$

Quine's method, as illustrated in Examples 1.3-1 and 1.3-2, becomes unwieldly for transmissions involving either many variables or many p-terms. This difficulty is overcome by simplifying the notation and making the procedure more systematic. The notation is simplified by discarding the expressions involving literals and using only the binary characters. This is permissible because the expressions in terms of literals can always be regained from the binary characters. The theorem being used to combine terms can be stated in terms of the binary characters as follows: If two binary characters are identical in all positions except one, and if neither character has a dash in the position in which they differ, then the two characters can be replaced by a single character which has a dash in the position in which the original characters differ and which is identical with the original characters in all other positions.

The first step in the revised method for determining prime implicants is to list in a column, such as that shown in Fig. 1.3-1a, the binary equivalents of the decimal numbers which specify the function. It is expedient to order these binary numbers so that any numbers which contain no 1's come first, followed by any numbers containing a single 1, etc. Lines should be drawn to divide the column into groups of binary numbers which contain a given number of 1's. The theorem stated above is applied to these binary numbers by comparing each number with all the numbers of the next lower group. Other pairs of numbers need not be considered since any two numbers which are not from adjacent groups must differ in more than one binary digit. For each number which has 1's wherever the number (from the next upper group) with which it is being compared has 1's, a new character is formed according to the theorem. A check mark is placed next to each number which is used in forming a new character. The new characters are placed in a separate column, such as Fig. 1.3-1b, I

	$x_5x_4x_3x_2x_1$	
0	0 0 0 0 0	✓
2	0 0 0 1 0	✓
4	0 0 1 0 0	✓
8	0 1 0 0 0	✓
16	1 0 0 0 0	✓
6	0 0 1 1 0	✓
10	0 1 0 1 0	✓
12	0 1 1 0 0	✓
18	1 0 0 1 0	✓
7	0 0 1 1 1	✓
11	0 1 0 1 1	✓
13	0 1 1 0 1	✓
14	0 1 1 1 0	✓
19	1 0 0 1 1	✓
29	1 1 1 0 1	✓
30	1 1 1 1 0	✓

(a)

II

	$x_5x_4x_3x_2x_1$	
0 2	0 0 0 - 0	✓
0 4	0 0 - 0 0	✓
0 8	0 - 0 0 0	✓
0 16	- 0 0 0 0	✓
2 6	0 0 - 1 0	✓
2 10	0 - 0 1 0	✓
2 18	- 0 0 1 0	✓
4 6	0 0 1 - 0	✓
4 12	0 - 1 0 0	✓
8 10	0 1 0 - 0	✓
8 12	0 1 - 0 0	✓
16 18	1 0 0 - 0	✓
6 7	0 0 1 1 -	
6 14	0 - 1 1 0	✓
10 11	0 1 0 1 -	
10 14	0 1 - 1 0	✓
12 13	0 1 1 0 -	
12 14	0 1 1 - 0	✓
18 19	1 0 0 1 -	
13 29	- 1 1 0 1	
14 30	- 1 1 1 0	

(b)

III

	$x_5x_4x_3x_2x_1$	
0 2 4 6	0 0 - - 0	✓
0 2 8 10	0 - 0 - 0	✓
0 2 16 18	- 0 0 - 0	
0 4 8 12	0 - - 0 0	✓
2 6 10 14	0 - - 1 0	✓
4 6 12 14	0 - 1 - 0	✓
8 10 12 14	0 1 - - 0	✓

(c)

IV

	$x_5x_4x_3x_2x_1$
0 2 4 6 8 10 12 14	0 - - - 0

(d)

Figure 1.3-1
Determination of Prime Implicants for Transmission
$T = \sum(0,2,4,6,7,8,10,11,12,13,14,16,18,19,29,30)$

which is again divided into groups of characters which have the same number of 1's. The characters in this new column will each contain one dash.

After each number in the first column has been considered, a similar process is carried out for the characters of column two. Two characters from adjacent groups can be combined if they both have their dashes in the same position and if the character from the lower group has 1's wherever the upper character has 1's. If any combinations are possible the resulting characters are placed in a third column such as Fig. 1.3-1c, and the Column II characters from which the new characters are formed are checked. All the characters in this third column will have two dashes. This procedure is repeated and new columns are formed (Fig. 1.3-1d) until no further combinations are possible. The unchecked characters, which have not entered into any combinations, represent the prime implicants.

Each binary character is labeled with the decimal equivalents of the binary numbers which it represents (see note in Example 1.3-1). These decimal numbers are arranged in increasing arithmetic order. For a character having one dash this corresponds to the order of its formation: When two binary numbers combine, the second number always contains all the 1's of the first number and one additional 1 so that the second number is always greater than the first. Characters having two dashes can be formed in two ways. For example, the character (12,13,14,15) can be formed either by combining (12,13) and (14,15) or by combining (12,14) and (13,15) as shown in Fig. 1.3-2.

0	0000
2	0010
4	0100
6	0110

02	00-0
04	0-00
26	0-10
46	01-0

0246	0--0
(0426	0--0)

Figure 1.3-2

Example of the Two Ways of

Forming a Character Having Two Dashes

Similarly, there are three ways in which a character having three dashes can be formed (in Fig. 1.3-1 the 0,2,4,6,8,10,12,14 character can be formed from the 0,2,4,6, and 8,10,12,14 characters or the 0,2,8,10, and 4,6,12,14 characters or the 0,4,8,12 and 2,6,10,14 characters), four ways in which a character having four dashes can be formed, etc.

In general, any character can be formed by combining two characters whose labels form an increasing sequence of decimal numbers when placed together. It is possible to shorten the process of determining prime implicants by not considering the combination of any characters whose labels do not satisfy this requirement. For example, in Fig. 1.3-1b the possibility of combining the (0,4) character with either the (2,6) or the (2,10) character need not be considered. If the process is so shortened, it is not sufficient to place check marks next to the two characters from which a new character is formed; each member of all pairs of characters which would produce the same new character when combined must also receive check marks. More simply, when a new character is formed a check mark is placed next to all characters whose labels contain only decimal numbers which occur in the label of the new character. In Fig. 1.3-1, when the (0,2,4,6) character is formed by combining the (0,2) and (4,6) characters, check marks must be placed next to the (0,4) and (2,6) characters as well as the (0,2) and (4,6) characters. If the process is not shortened as just described, the fact that a character can be formed in several ways can serve as a check on the accuracy of the process.

1.4 Prime Implicant Tables

The minimum sum is formed by picking the fewest prime implicants whose sum will equal one for all rows of the table of combinations for which the transmission is to equal one. In terms of the characters used in Section 1.3 this means that each number in the decimal specification of the function must appear in the label of at least one character which corresponds to a ms-term (term of the minimum sum).

The ms-terms are selected from the prime implicants by means of a prime implicant table* (Fig. 1.4-1). Each column of the prime implicant table corresponds to a row of the table

	0	2	4	8	16	6	10	12	18	7	11	13	14	19	29	30	
1	x	x	x	x		x	x	x					x				*
2	x	x			x				x								*
3													x			x	*
4												x			x		*
5									x					x			*
6								x				x					
7							x				x						*
8						x				x							*

Figure 1.4-1

Prime Implicant Table for the Transmission of Fig. 1.3-1

---

* This table was first discussed by Quine [Q1]. However, no systematic procedure for obtaining a minimum sum from the prime implicant table was presented.

of combinations for which the transmission is to have the value one. The decimal number at the top of each column specifies the corresponding row of the table of combinations. Thus the numbers which appear at the tops of the columns are the same as those which specify the transmission. Each row of the prime implicant table represents a prime implicant. If a prime implicant equals "one" for a given row of the table of combinations, a cross is placed at the intersection of the corresponding row and column of the prime implicant table. All other positions are left blank. The table can be written directly from the characters obtained in Section 1.3 by identifying each row of the table with a character and then placing a cross in each column whose number appears in the label of the character.

It is convenient to arrange the rows in the order of the number of crosses they contain, with those rows containing the most crosses at the top of the table. Also, horizontal lines should be drawn partitioning the table into groups of rows which contain the same number of crosses (Fig. 1.4-1). If, in selecting the rows which are to correspond to ms-terms, a choice between two equally appropriate rows is required, the row having more crosses should be selected. The row with more crosses has fewer literals in the corresponding prime implicant. This choice is more obvious when the table is partitioned as suggested above.

A minimum sum is determined from the prime implicant table by selecting the fewest rows such that each column has a cross in at least one selected row. The selected rows are called basis rows, and the prime implicants corresponding to the basis rows are the ms-terms. If any column has only one entry, the row in which this entry occurs must be a basis row. Therefore the first step in selecting the basis rows is to place an asterisk next to each row which contains the sole entry of any column (rows 1,2,3,4,5,7,8, in Fig. 1.4-1). A line is then drawn through all rows marked with an asterisk and through all columns in which these rows have entries. This is done because the requirement that these columns have entries in at least one basis row is satisfied by selecting the rows marked with an asterisk as basis rows. When this is done for the table of Fig. 1.4-1 all columns are lined out and therefore the rows marked with asterisks are the basis rows for this table. Since no alternative choice of basis rows is possible, there is only one minimum sum form for the transmission described in this table.

1.5 Row Covering

In general, after the appropriate rows have been marked with asterisks and the corresponding columns have been lined out, there may remain some columns which are not lined out; for example, column 7 in the table of Fig. 1.5-1b. When this happens, additional rows must be selected and the columns in which these rows have entries must be lined out until all columns of the table are lined out. For the table of Fig. 1.5-1b, the selection of either row 2 or row 6 as a basis row will cause column 7 to be lined out. However, row 2 is the correct choice since it has more crosses than row 6. This is an example of the situation which was described earlier in connection with the partitioning of prime implicant tables. Row 2 is marked with two asterisks to indicate that it is a basis row even though it does not contain the sole entry of any column.

The choice of basis rows to supplement the single asterisk rows becomes more complicated when several columns (such as columns 2,3, and 6 in Fig. 1.5-2a) remain to be lined out. The first step in choosing these supplementary basis rows is to determine whether any pairs of rows exist such that one row has crosses only in columns in which the other member of the pair has crosses. Crosses in lined-out columns are not considered. In Fig. 1.5-2a, rows 1 and 2 and rows 2 and 3

$T = \Sigma(0,1,2,3,7,14,15,22,23,29,31)$

	x_5	x_4	x_3	x_2	x_1	
0	0	0	0	0	0	✓
1	0	0	0	0	1	✓
2	0	0	0	1	0	✓
3	0	0	0	1	1	✓
7	0	0	1	1	1	✓
14	0	1	1	1	0	✓
22	1	0	1	1	0	✓
15	0	1	1	1	1	✓
23	1	0	1	1	1	✓
29	1	1	1	0	1	✓
31	1	1	1	1	1	✓

	x_5	x_4	x_3	x_2	x_1	
0 1	0	0	0	0	-	✓
0 2	0	0	0	-	0	✓
1 3	0	0	0	-	1	✓
2 3	0	0	0	1	-	✓
3 7	0	0	-	1	1	
7 15	0	-	1	1	1	✓
7 23	-	0	1	1	1	✓
14 15	0	1	1	1	-	
22 23	1	0	1	1	-	
15 31	-	1	1	1	1	✓
23 31	1	-	1	1	1	✓
29 31	1	1	1	-	1	

	x_5	x_4	x_3	x_2	x_1
0 1 2 3	0	0	0	-	-
7 15 23 31	-	-	1	1	1

(a) Determination of Prime Implicants

$T = \Sigma[(0,1,2,3),$ $(7,15,23,31),$ $(29,31),$ $(22,23),$ $(14,15)]$

$T = x_5'x_4'x_3' + x_3x_2x_1 + x_5x_4x_3x_1 + x_5x_4'x_3x_2 + x_5'x_4x_3x_2$

(c) Minimum Sum

	0	1	2	3	7	14	22	15	23	29	31	
1	x	x	x	x								*
2					x			x	x		x	* *
3										x	x	*
4							x		x			*
5						x		x				*
6				x	x							

(b) First Step in Selection of Basis Rows

Figure 1.5-1

Determination of the Minimum Sum for $T = \Sigma(0,1,2,3,7,14,15,22,23,29,31)$

	0	1	2	64	3	6	33	7	14	22	30	71	78	86	62	
1						x			x	x	x					
2			x		x	x		x								
3	x	x			x											
4											x				x	*
5										x				x		*
6									x				x			*
7								x				x				*
8		x					x									*
9	x			x												*

(a) Prime Implicant Table with Single Asterisk Rows and Corresponding Columns Lined Out

	0	1	2	64	3	6	33	7	14	22	30	71	78	86	62	
1						x			x	x	x					
2			x		x	x		x								**
3	x	x	x		x											
4											x				x	*
5										x				x		*
6									x				x			*
7								x				x				*
8		x					x									*
9	x			x												*

(b) Prime Implicant Table with Rows Which are Covered by Other Rows Lined Out

Figure 1.5-2

Prime Implicant Tables for

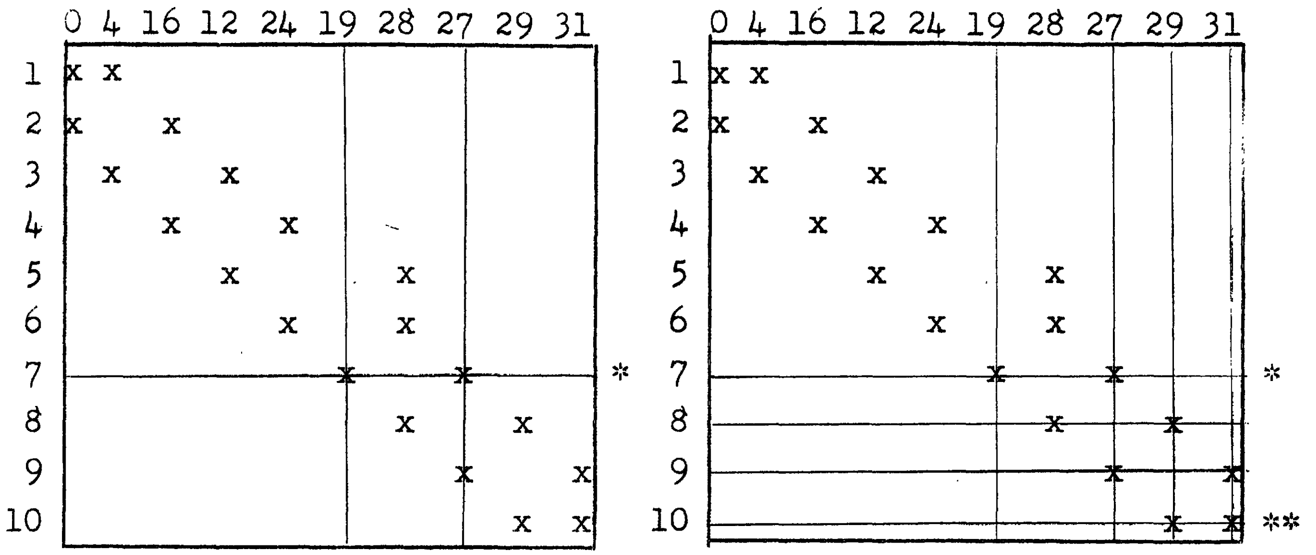
$T = \Sigma(0,1,2,3,6,7,14,22,30,33,62,64,71,78,86)$

are such pairs of rows since row 2 has crosses in columns 2,3, and 6 and row 1 has a cross in column 6 and row 3 has crosses in columns 2 and 3. A convenient way to describe this situation is to say that row 2 covers rows 1 and 3, and to write $2 \supset 1$, $2 \supset 3$. If row i is selected as a supplementary basis row and row i is covered by row j, which has the same total number of crosses as row i, then it is possible to choose row j as a basis row instead of row i since row j has a cross in each column in which row i has a cross.

The next step is to line out any rows which are covered by other rows in the same partition of the table (rows 1 and 3 in Fig. 1.5-2a). If any column now contains only one cross which is not lined out (columns 2 and 3, and 6 in Fig. 1.5-2b), two asterisks are placed next to the row in which this cross occurs (row 2 in Fig. 1.5-2b) and this row and all columns in which this row has crosses are lined out. The process of drawing a line through any row which is covered by another row and selecting each row which contains the only cross in a column is continued until it terminates. Either all columns will be lined out in which case the rows marked with one or two asterisks are the basis rows, or each column will contain more than one cross and no row will cover another row. The latter situation is discussed in the following section.

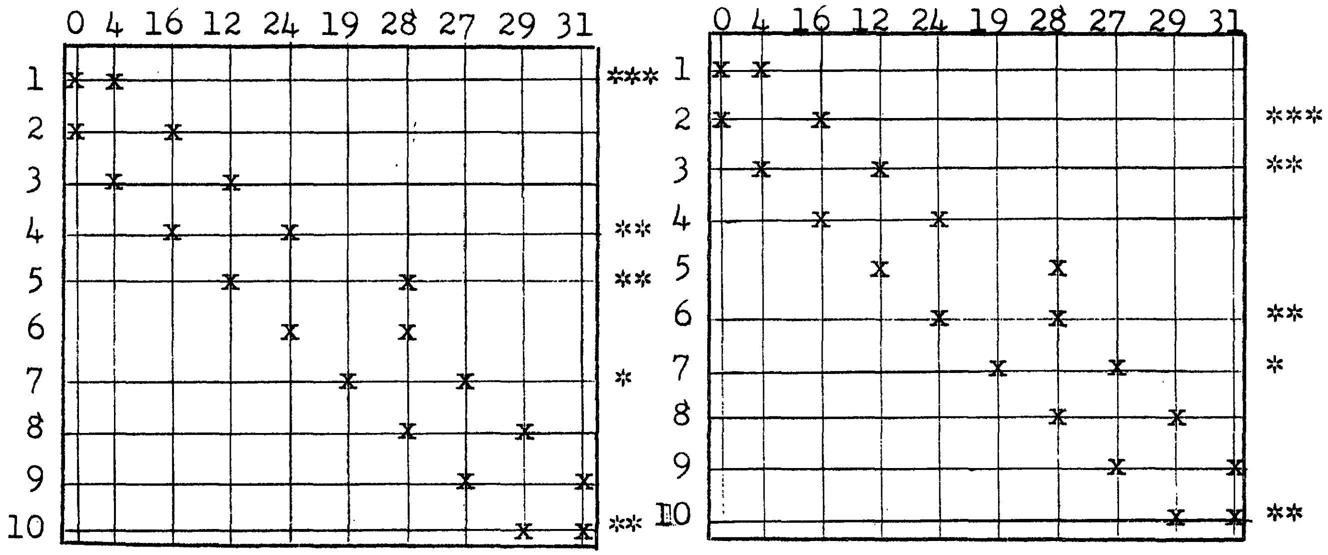
1.6 Prime Implicant Tables in Cyclic Form

If the rows and columns of a table which are not lined out are such that every column has more than one cross and no row covers another row (Fig. 1.6-1b), the table will be said to be in cyclic form, or, in short, to be cyclic. If any column has crosses in only two rows (in Fig. 1.6-1b, column 0 has crosses in rows 1 and 2) at least one of these rows must be included in any set of basis rows. Therefore, the basis rows for a cyclic table can be discovered by first determining whether any column contains only two crosses and if such a column exists, by then selecting as a trial basis row one of the rows in which the crosses of this column occur. If no column contains only two crosses, then a column which contains three crosses is selected, etc. All columns in which the trial basis row has crosses are lined out and the process of lining out rows which are covered by other rows and selecting each row which contains the only cross of some column is carried out as described above. Either all columns will be lined out or another cyclic table will result. Whenever a cyclic table occurs, another trial row must be selected. Eventually all columns will be lined out. However, there is no guarantee that the selected rows are actually basis rows. The possibility exixts that a different choice of trial rows would have resulted in fewer selected rows. In



(a) Selection of Single Asterisk Rows

(b) Selection of Double Asterisk Rows



(c) Selection of Row 1 as a Trial Basis Row (Col. 0)

(d) Selection of Row 2 as a Trial Basis Row (Col. 0)

Figure 1.6-1
Determination of Basis Rows for a
Cyclic Prime Implicant Table

general, it is necessary to carry out the procedure of selecting rows several times, choosing different trial rows each time, so that all possible combinations of trial rows are considered. The set of fewest selected rows is the actual set of basis rows.

Figure 1.6-1 illustrates the process of determining basis rows for a cyclic prime implicant table. After rows 7 and 10 have been selected a cyclic table results (Fig. 1.6-1b). Rows 1 and 2 are then chosen as a pair of trial basis rows since column 0 has crosses in only these two rows. The selection of row 1 leads to the selection of rows 4 and 5 as shown in Fig. 1.6-1c. Row 1 is marked with three asterisks to indicate that it is a trial basis row. Figure 1.6-1d illustrates the fact that the selection of rows 3 and 6 is brought about by the selection of row 2. Since both sets of selected rows have the same number of rows (5) they are both sets of basis rows. Each set of basis rows corresponds to a different minimum sum so that there are two minimum sum forms for this function.

Sometimes it is not necessary to determine all minimum sum forms for the transmission being considered. In such cases, it may be possible to shorten the process of determining basis rows. Since each column must have a cross in some basis row, the total number of crosses in all of the basis rows is equal to or greater than the number of columns. Therefore, the number of columns divided by the greatest number of crosses in any row (or the next highest integer if this ratio is not an integer) is equal to the fewest possible basis rows. For example, in the table of Fig. 1.6-1 there are ten columns and two crosses in each row. Therefore, there must be at least 10 divided by 2 or 5 rows in any set of basis rows. The fact that there are only five rows selected in Fig. 1.6-1c guarantees that the selected rows are basis rows and therefore Fig. 1.6-1d is unnecessary if only one minimum sum is required. In general, the process of trying different combinations of trial rows can be stopped as soon as a set of selected rows which contains the fewest possible number of basis rows has been found (providing that it is not necessary to discover <u>all</u> minimum sum forms). It should be pointed out that more than the minimum number of basis rows may be required in some cases and in these cases all combinations of trial rows must be considered. A more accurate lower bound on the number of basis rows can be obtained by considering the number of rows which have the most crosses. For example, in the table of Fig. 1.5-2 there are 15 columns and 4 crosses, at most, in any row. A lower bound of 4 (15/4 = 3 3/4) is a little too optimistic since there are only three rows which contain four crosses. A more realistic lower bound of 5 is obtained by noting that the rows which have 4 crosses can provide crosses in at most 12 columns and that at least two additional rows containing two crosses are necessary to provide crosses in the three remaining columns.

1.7 Cyclic Prime Implicant Tables and Group Invariance

It is not always necessary to resort to enumeration in order to determine all minimum sum forms for a cyclic prime implicant table. Often there is a simple relation among the various minimum sums for a transmission so that they can all be determined directly from any single minimum sum by simple interchanges of variables. The process of selecting basis rows for a cyclic table can be shortened by detecting beforehand that the minimum sums are so related.

An example of a transmission for which this is true is shown in Fig. 1.7-1. If the variables x_1 and x_2 are interchanged, one of the minimum sums is changed into the other. In the prime implicant table the interchange of x_1 and x_2 leads to the interchange of columns 1 and 2, 5 and 6, 9 and 10, 13 and 14, and rows 1 and 2, 3 and 4, 5 and 6, 7 and 8. The transmission itself remains the same after the interchange.

In determining the basis rows for the prime implicant table (Fig. 1.7-1(d)) either row 1 or row 2 can be chosen as a trial basis row. If row 1 is selected the i-set of basis rows will result and if row 2 is selected the ii-set of basis rows will result. It is unnecessary to carry out the procedure of determining both sets of basis rows. Once the i-set of basis rows is known, the ii-set can be determined directly by interchanging the x_1 and x_2 variables in the i-set. Thus no enumeration is necessary in order to determine all minimum sums.

	x_4	x_3	x_2	x_1	
0	0	0	0	0	✓
1	0	0	0	1	✓
2	0	0	1	0	✓
5	0	1	0	1	✓
6	0	1	1	0	✓
9	1	0	0	1	✓
10	1	0	1	0	✓
7	0	1	1	1	✓
11	1	0	1	1	✓
13	1	1	0	1	✓
14	1	1	1	0	✓
15	1	1	1	1	✓

(a)

	x_4	x_3	x_2	x_1	
0, 1	0	0	0	-	
0, 2	0	0	-	0	
1, 5	0	-	0	1	✓
1, 9	-	0	0	1	✓
2, 6	0	-	1	0	✓
2,10	-	0	1	0	✓
5, 7	0	1	-	1	✓
5,13	-	1	0	1	✓
6, 7	0	1	1	-	✓
6,14	-	1	1	0	✓
9,11	1	0	-	1	✓
9,13	1	-	0	1	✓
10,11	1	0	1	-	✓
10,14	1	-	1	0	✓
7,15	-	1	1	1	✓
11,15	1	-	1	1	✓
13,15	1	1	-	1	✓
14,15	1	1	1	-	✓

(b)

	x_4	x_3	x_2	x_1
1, 5, 9,13	-	-	0	1
2, 6,10,14	-	-	1	0
5, 7,13,15	-	1	-	1
6, 7,14,15	-	1	1	-
9,11,13,15	1	-	-	1
10,11,14,15	1	-	1	-

(c)

0	1	2	5	6	9	10	7	11	13	14	15
	x		x		x				x		
		x		x		x				x	
			x				x		x		x
				x			x			x	x
					x			x	x		x
						x		x		x	x
x	x										
x		x									

(d)

(i) (0,1)+(2,6,10,14)+(5,7,13,15)+(9,11,13,15)

(ii) (0,2)+(1,5, 9,13)+(6,7,14,15)+(10,11,14,15)

$$T_i = x_4'x_3'x_2' + x_2x_1' + x_3x_1 + x_4x_1$$

$$T_{ii} = x_4'x_3'x_1' + x_1x_2' + x_3x_2 + x_4x_2$$

(e)

Figure 1.7-1
Determination of the Minimum Sums for
$T = \sum(0,1,2,5,6,7,9,10,11,13,14,15)$

In general, the procedure for a complex prime implicant table is to determine whether there are any pairs of variables which can be interchanged without effecting the transmission. If such pairs of variables exist, the corresponding interchanges of pairs of rows are determined. A trial basis row is then selected from a pair of rows which contain the only two crosses of a column and which are interchanged when the variables are permuted. After the set of basis rows has been determined, the other set of basis rows can be obtained by replacing each basis row by the row with which it is interchanged when variables are permuted. If any step of this procedure is not possible, it is necessary to resort to enumeration.

In the preceding discussion only simple interchanges of variables have been mentioned. Actually all possible permutations of the contact variables should be considered. It is also possible that priming variables or both priming and permuting them will leave the transmission unchanged. For example, if $T = x_4 x_3' x_2 x_1' + x_4' x_3 x_2' x_1$, priming all the variables leaves the function unchanged. Also, priming x_4 and x_3 and then interchanging x_4 and x_3 does not change the transmission. The general name for this property is <u>group invariance</u>. This was discussed by Shannon in [S2]. A method for determining the group invariance for a specified transmission will be presented in Appendix A.

1.8 ϕ-Terms

In section 1.1 the possibility of having ϕ-entries in a table of combinations was mentioned. Whenever there are combinations of the relay conditions for which the transmission is not specified, ϕ-entries are placed in the T-column of the corresponding rows of the table of combinations. The actual values (0 or 1) of these ϕ-entries are chosen so as to simplify the form of the transmission. This section will describe how to modify the method for obtaining a minimum sum when the table of combinations contains ϕ-entries.

The p-terms which correspond to ϕ-entries in the table of combinations will be called ϕ-terms. These ϕ-terms should be included in the list of p-terms which are used to form the prime implicants. See Fig. 1.8-1a. However, in forming the prime implicant table, columns corresponding to the ϕ-terms should <u>not</u> be included (Fig. 1.8-1b). The ϕ-terms are used in forming the prime implicants in order to obtain prime implicants containing the fewest possible literals. If columns corresponding to the ϕ-terms were included in forming the prime implicant table this would correspond to setting all the ϕ-entries in the table of combinations equal to 1. This does not necessarily lead to the simplest minimum sum. In the procedure just described, the ϕ-entries will automatically be set equal to either 0 or 1 so as to produce the simplest minimum sum. For the transmission of Fig. 1.8-1 the 14 ϕ-entry has been set equal to 1 and the 9 ϕ-entry has been set equal to 0.

	x_4	x_3	x_2	x_1	
5	0	1	0	1	✓
6	0	1	1	0	✓
(ϕ) 9	1	0	0	1	✓
13	1	1	0	1	✓
(ϕ)14	1	1	1	0	✓

	x_4	x_3	x_2	x_1
5, 13	-	1	0	1
6, 14	-	1	1	0
9, 13	1	-	0	1

(a) Determination of Prime Implicants

	5	6	13
*	x		x
*		x	
			x

(b) Prime Implicant Table

(c) Basis rows: (5,13), (6,14) (d) $T = x_3x_2'x_1 + x_3x_2x_1'$

Fig. 1.8-1 Determination of the Minimum Sum for the Transmission $T = \Sigma(5,6,13) + \phi(9,14)$ where 9 and 14 are the ϕ-terms.

This section concludes Part 1. A method has been presented for determining the minimum sums which correspond to any transmission specified by a table of combinations. The possibility that the transmission may not be specified for certain rows of the table of combinations has been taken into account. The minimum sum will be used as the starting point for the design procedure for contact networks to be presented in Part 2.

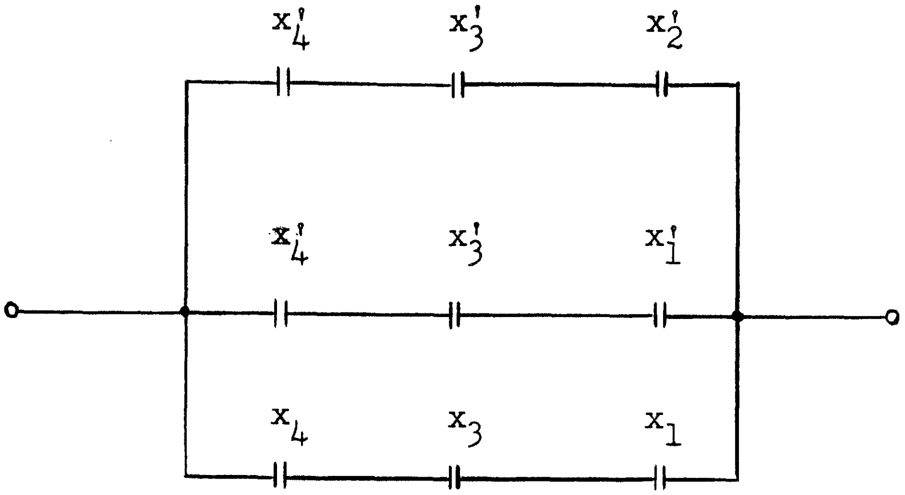
PART 2

THE DESIGN OF TWO-TERMINAL CONTACT NETWORKS

2.1 History of the Problem

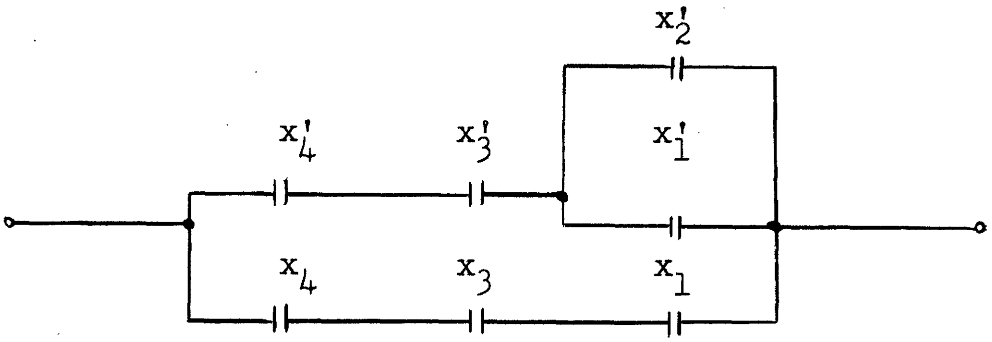
Usually, the first step in designing a two-terminal contact network is to express the desired transmission as a minimum sum. This can always be done by means of the procedure presented in Part 1. A contact network whose transmission equals this minimum sum can then be designed by applying backwards the method given in Section 1.1 for analyzing series-parallel networks. This will always result in a network which consists of several branches in parallel. Each branch will correspond to a ms-term (term of the minimum sum) and will be a series connection of contacts. For example, the minimum sum for the transmission $T = \sum(0,1,2,13,14)$ is $T = x_4'x_3'x_2' + x_4'x_3'x_1' + x_4x_3x_1$, which corresponds to the network shown in Fig. 2.1-1a.

A network designed directly from a minimum sum is not generally desirable since such a network usually contains more contacts than necessary. It is very often possible to factor a minimum sum before designing the network and thereby obtain a network with fewer contacts. Fig. 2.1-1b shows the network which results when the transmission of Fig. 2.1-1a is factored (note that two contacts are saved). One difficulty with the method of factorization is that no definite rules can be given for how to proceed. For example, the transmission $T = \sum(0,1,2,3,4,11)$, is written as the minimum sum, $T = x_4'x_3' + x_4'x_2'x_1' + x_3'x_2x_1$,



(a) Without Factoring (nine contacts)

$$T = x_4'x_3'x_2' + x_4'x_3'x_1' + x_4x_3x_1$$



(b) With Factoring (seven contacts)

$$T = x_4'x_3' (x_2' + x_1') + x_4x_3x_1$$

Figure 2.1-1

Network for Transmission

$T = \Sigma(0,1,2,13,14)$

and factored as follows: $T = x_4'x_3' + x_4'x_2'x_1' + x_3'x_2x_1 =$ $x_4' (x_3' + x_2'x_1') + x_2x_1x_3' + x_2x_1x_2'x_1' = x_4' (x_3' + x_2'x_1') +$ $x_2x_1 (x_3' + x_2'x_1') = (x_4' + x_2x_1) (x_3' + x_2'x_1')$. The second difficulty with the method of factorization is that it restricts the final network to be a series-parallel network. Very frequently, bridge or non-planar networks use many fewer contacts than any series-parallel network having the same transmission. Any method which designs only series-parallel networks must be rejected if economy of contacts is important.

In 1949, Shannon [S2] presented a design method which does not necessarily result in a series-parallel network. In this method, the desired two-terminal network is formed by connecting together two multi-terminal networks. See Fig. 2.1-2. One of these multi-terminal networks, called the M-network, is always a contact tree [C1], [KRW1]. (A contact tree on n-variables is a network which has one input and 2^n outputs; the transmission between the input and each output is equal to one of the p-terms of the n-variables.) The second multi-terminal network, called the N-network, is not of any special form.

One difficulty with this design method is the fact that it is usually not clear which variables should be included in the M-network. Several possibilities must usually be tested. Another difficulty is that the economy of the design often

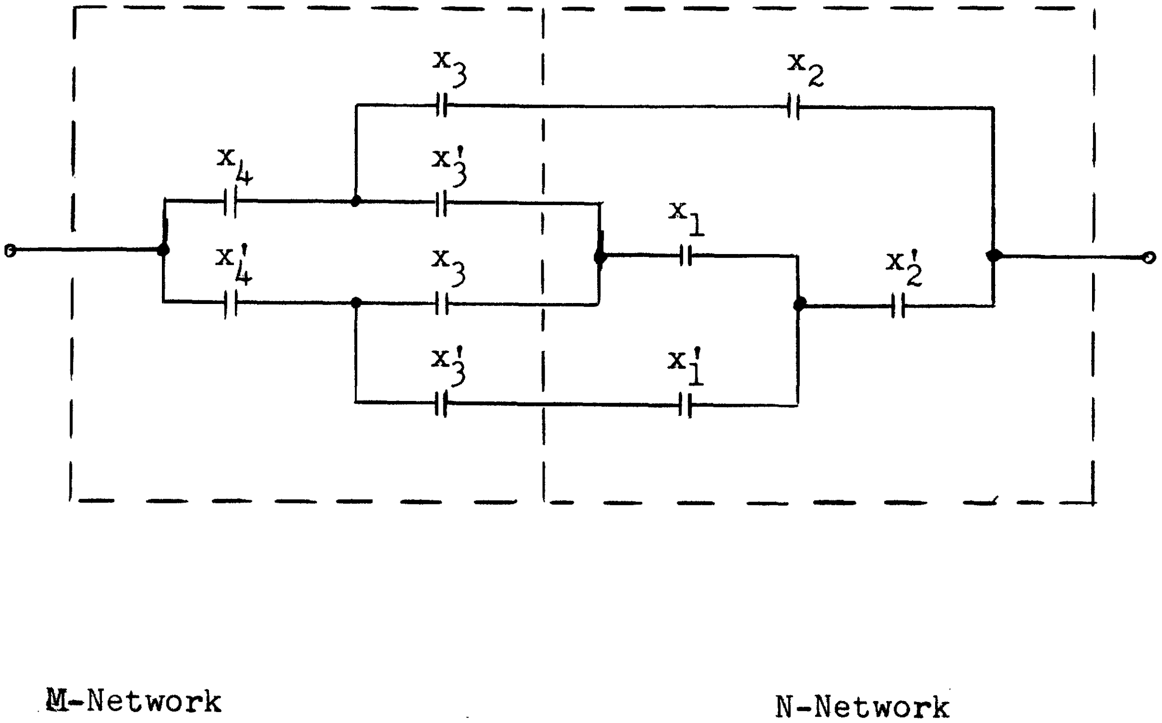


Figure 2.1-2

Network designed by Shannon M-N Method

Transmission is $T = \Sigma(0,5,9,14,15)$

depends on the way in which the N-network is modified to eliminate contacts. No general rules for these modifications have yet been formulated. The major difficulty with the M-N design is the fact that it assumes that part of the network will be a contact tree. Many transmissions can be realized most economically by networks which do not contain contact trees and hence the M-N design procedure must be rejected as a general method for designing economical contact networks.

Finally, the matrix method of Hohn and Schissler [HS1] is truly general since no special form of network is assumed. However, no definite rules of procedure are given so that the success of the method depends on the skill of the designer.

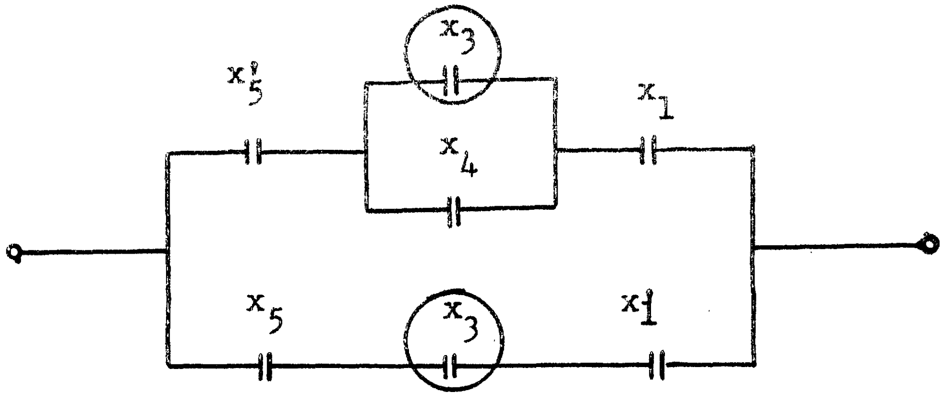
All existing methods of contact network design are unsatisfactory either because they restrict the network to be of some special form or else because they depend on the skill of the designer.

2.2 General Principles - Essential Contacts

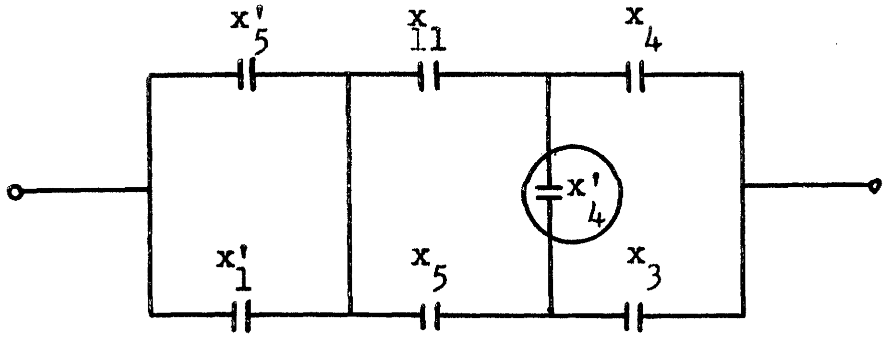
The design method which is to be presented in the following sections is completely general in the sense that no special form of network is assumed. Moreover, this method is more systematic than the only other general design method available, the matrix method.

The desired transmission is first written as a minimum sum. The network is then formed by providing paths which correspond to the ms-terms between the external terminals. A path between the external terminals of a network will be called a total path. The transmission of each total path is made equal to one of the ms-terms. This design procedure corresponds to the second analysis method (path tracing) which was presented in Section 1.1

It is not necessary that the total paths correspond to ms-terms. In certain situations, it will be desirable to substitute different product terms for some of the ms-terms (Fig. 2.2-1b). These substitutions are discussed in Section 2.9. However, the minimum sum is the most reasonable starting point since it requires the fewest paths (each additional path usually necessitates the addition of at least one contact). Also, it is easier to go from a minimum sum to a non-minimum sum than it is to go from one non-minimum sum to a different non-minimum sum. Until it is possible to know beforehand when a non-minimum sum is appropriate and just which non-minimum sum should be used, a minimum sum will be the best form to start with.



(a) $T = x_5'x_1x_3 + x_5'x_1x_4 + x_5x_3x_1'$. Two contacts are labeled with the essential literal x_3. 7 contacts, 12 springs



(b) $T = x_5'x_1x_4 + x_5'x_1x_3 + x_5x_3x_1'$. The non-essential literal x_4' occurs as a contact designator. 7 contacts, 11 springs

(c) Minimum Sum: $T = x_5'x_1x_3 + x_5'x_1x_4 + x_5x_3x_1'$

(d) Essential Literals: $x_5, x_5', x_4, x_3, x_1, x_1'$

Figure 2.2-1

Networks for transmission $T = \Sigma(3,5,7,10,14)$ showing non-essential and duplicate contacts

Each literal which appears in a minimum sum must also occur as a contact designator in any network which has a transmission equivalent to the minimum sum. (In forming the minimum sum, all literals which could possibly be eliminated were eliminated.) The literals which occur in the minimum sum will be called essential literals, and the remaining literals will be called non-essential literals.

Any network in which each essential literal appears as the designator of only one contact and no non-essential literal appears as a contact designator is optimum in the sense that it contains the fewest possible contacts. Such a network is not always possible. At present, the use of transfer contacts to reduce the total number of springs will not be considered. See [C1], Section 4.10.

Any contact which has a non-essential literal as its designator will be called a non-essential contact (x_4' contact in Fig. 2.2-1b). Contacts designated by essential literals will be called essential contacts. If more than one contact is designated by the same literal it will be said that there are duplicate contacts present. It is not possible to specify which of the contacts are the duplicate contacts; all that can be done is to count the total number (k) of contacts designated by the same literal (x^*) subtract one from this number (k-1), and then say that there are k-1 duplicate x^* contacts. For example, in the network of Fig. 2.2-1a there is one duplicate x_3 contact. The network of Fig. 2.2-1b, which contains one non-essential contact, has been proven to contain the fewest possible contacts and springs [M1].

2.3 Outline of Design Procedure

The first step in the design procedure discussed here is to form a network whose transmission equals the sum of two ms-terms of the desired transmission. Contacts are then added to this network so as to form a total path (a path between the input and output terminals) which corresponds to another ms-term of the desired transmission. The process of adding contacts is repeated until a network whose transmission equals the desired transmission is formed.

A transmission which is the sum of two ms-terms can always be realized by a network which contains no non-essential or duplicate contacts. Such a transmission can always be written in the form: $T = x_1^* x_2^* \cdots x_i^* (x_j^* x_{j+1}^* \cdots x_s^* + x_k^* x_{k+1}^* \cdots x_t^*)$ where $x_1^*, x_2^*, \cdots x_i^*$ represent the literals which are common to both ms-terms. Fig. 2.3-1 shows the general form of such a network.

The design procedure is completely specified when the method of adding contacts to form a path corresponding to a ms-term is described and a rule is given for determining the order in which the ms-terms are to be selected. The question of order is considered in Section 2.10. The formation of paths corresponding to ms-terms will be discussed next.

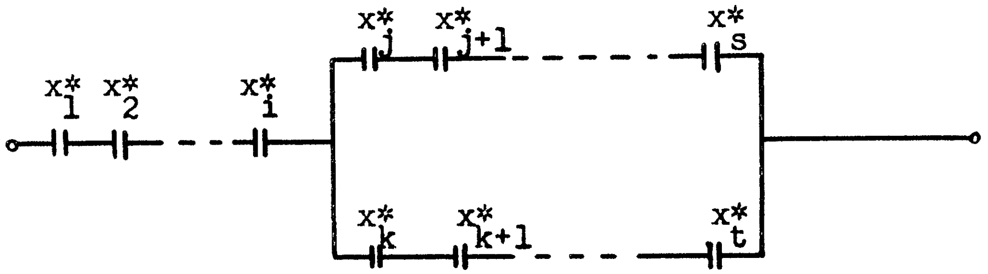


Figure 2.3-1

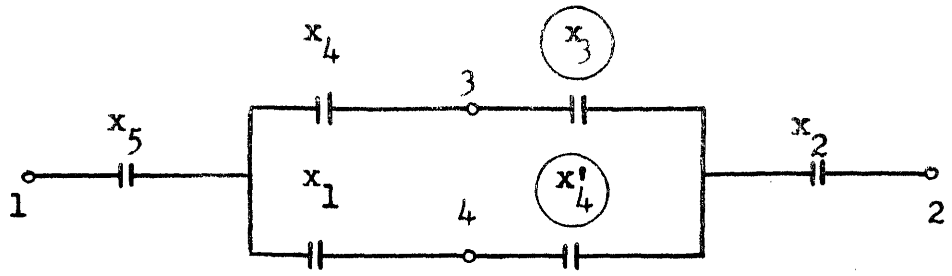
General Form of Network whose Transmission is the Sum of Two ms-terms

2.4 The Addition of Paths to a Network - Definition of "Equal Add T"

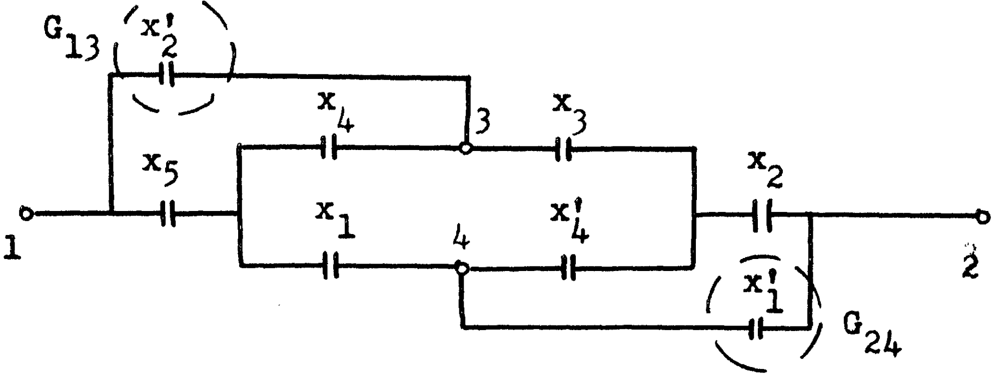
In order to obtain a network which uses as few contacts as possible, it is necessary to avoid adding any non-essential or duplicate contacts when forming total paths. This means that an attempt should be made to include in the path being formed any contacts in the network whose designators are the same as literals in the ms-term corresponding to this path. It will be necessary to add a contact for each literal of the ms-term which does not occur as a contact designator.

The first step in adding a ms-term path is to circle all designators which are the same as literals in the ms-term. A contact whose designator is circled will be called a desired contact and a series connection of desired contacts will be called a desired path. In Fig. 2.4-1a, the x_3 and x_4' contacts form a desired path. The procedure for a single desired path will be described next. Multiple desired paths and parallel connections of desired contacts will be considered in Sections 2.8 and 2.9.

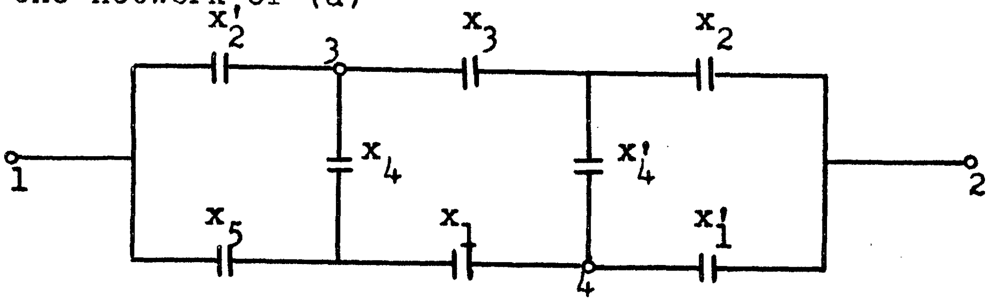
Denote the input and output nodes of the network by 1 and 2, respectively, and the first and last nodes of the desired path by 3 and 4. See Fig. 2.4-1a. Let M_i be the ms-term for which a path is to be added, and T the transmission for which the network is to be designed $(T = \sum M_i)$. T_{12} will



(a) Network for Transmission $x_5x_4x_3x_2+ x_5x'_4x_2x_1$ with designators which occur in $x'_4x_3x'_2x'_1$ circled



(b) Network having transmission $x_5x_4x_3x_2+ x_5x'_4x_2x_1+ x'_4x_3x'_2x'_1$ formed by adding contacts (x'_1 and x'_2) to the network of (a)



(c) Network of (b) redrawn

Figure 2.4-1

Design of a contact network having transmission

$T =\Sigma(4,19,20,23,30,31) = x_5x_4x_3x_2+ x_5x'_4x_2x_1+ x'_4x_3x'_2x'_1$

represent the total transmission between terminals 1 and 2 (the sum of the transmissions of all paths joining terminals 1 and 2), T_{34} will represent the total transmission between nodes 3 and 4, etc.

Contacts are to be added to the network so that a path between terminals 1 and 2 having transmission M_i is formed. In order to avoid duplicate contacts, it is necessary to include the contacts of the desired path in this path. This is done by adding a series connection of contacts having transmission G_{13} (G_{14}) between terminals 1 and 3 (1 and 4) and a series connection of contacts having transmission G_{24} (G_{23}) between terminals 2 and 4 (2 and 3). The transmissions G_{13} and G_{24} (or G_{14} and G_{23}) must be chosen so that:

(i) $G_{13}\ G_{24}\ T_{34} = M_i$ $\qquad$ $G_{14}\ G_{23}\ T_{34} = M_i$

(iia) $G_{13}\ T_{23} = 0$ $\qquad$ or $\qquad$ $G_{14}\ T_{24} = 0$

(iib) $G_{24}\ T_{14} = 0$ $\qquad$ $G_{23}\ T_{13} = 0$

Equation (i) requires that the path formed by adding G_{13} and G_{24} (or G_{14} and G_{23}) has a transmission equal to M_i and equations (ii) require that no unwanted (or "sneak") paths are formed. For the network in Fig. 2.4-1a, $T_{23} = x_2x_3$, $T_{14} = x_5x_1$, $T_{34} = x_1x_4 + x_3x_4'$. If x_2' and x_1' are chosen as G_{13} and G_{24}, respectively, the circuit of Fig. 2.4-1b results and equations (i) and (ii) become:

(i) $G_{13}\ G_{24}\ T_{34} = x_2'\ x_1'\ (x_1x_4 + x_3x_4') = x_1'x_2'x_3x_4' = M_3$

(iia) $G_{13}\ T_{23} = x_2'\ (x_2x_3) = 0$

(iib) $G_{24}\ T_{14} = x_1'\ (x_5x_1) = 0$

as required.

Fig. 2.4-2 shows another example of a network designed by adding G_{14} and G_{23} transmissions. For this network $T_{13} = x_3x_4$, $T_{24} = x_1$, $T_{34} = x_1x_2 + x_2'x_3$, $M_3 = x_5x_3x_2'x_1'$, $G_{14} = x_1'$, $G_{23} = x_5$, and:

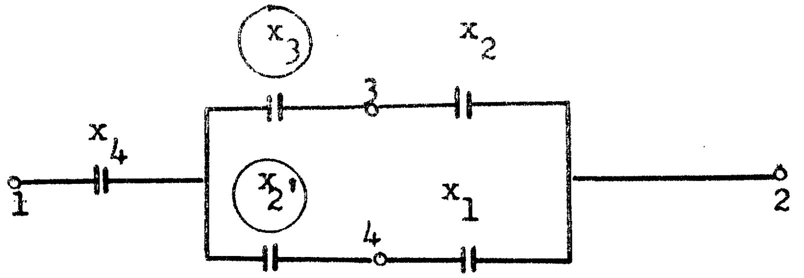
(i) $G_{14}\ G_{23}\ T_{34} = x_1'x_5\ (x_1x_2 + x_2'\ x_3) = x_1'x_2'x_3x_5 = M_3$

(iia) $G_{14}\ T_{42} = (x_1')\ (x_1) = 0$

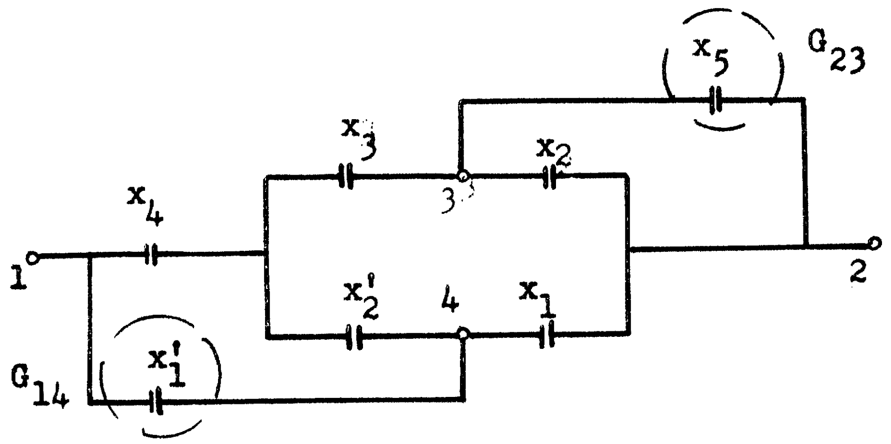
(iib) $G_{23}\ T_{31} = (x_5)\ (x_3x_4) = \Sigma(28,29,30,31)$

Even though equation (iib) is not satisfied, the network does have the desired transmission since the "sneak" path has transmission $x_5x_4x_3 = \Sigma(28,29,30,31)$ which equals one only when the desired transmission, $T = \Sigma(9,13,14,15,20,25,28,29,30,31)$, equals one.

This example points out the fact that equations (i) and (ii) are more restrictive than necessary. As stated, these equations do not allow any "sneak" paths. Actually, only "sneak" paths which have transmissions which equal one when the desired over-all transmission equals zero must be excluded. If such paths were present, they would connect the external terminals together in circuit conditions for which it is desired to have the external terminals separated.



(a) Network for transmission $x_4(x_2x_3 + x_2'x_1)$ with designators which occur in $x_5x_3x_2'x_1'$ circled



(b) Network having transmission $x_5x_3x_2'x_1' + x_4x_2'x_1 + x_4x_3x_2$ formed by adding $G_{14} = x_1'$ and $G_{23} = x_5$ to the network of (a)

Figure 2.4-2

Design of a Contact Network having Transmission $T = \Sigma(9,13,14,15,20,25,28,29,30,31) = x_5x_3x_2'x_1' + x_4x_2'x_1 + x_4x_3x_2$

Equations (i) and (ii) can be made less restrictive by allowing "sneak" paths whose transmissions equal one only when the desired transmission equals one. Algebraically, this means that the left-hand side of (i) can be M_i plus any p-terms contained in T and the left-hand sides of equations (ii) can be 0 plus any p-terms contained in T. This will be written symbolically as:

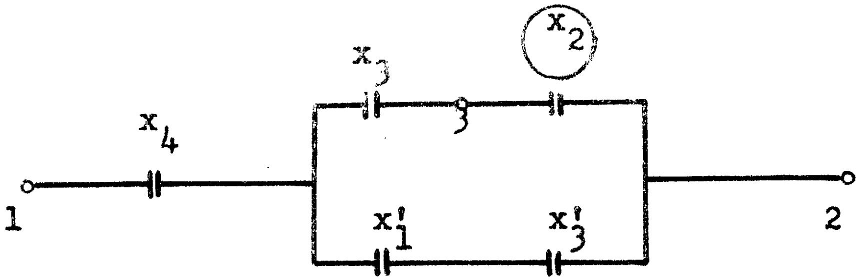
(i')	$G_{13}\ G_{24}\ T_{34} = M_i\ (aT)$		$G_{14}\ G_{23}\ T_{34} = M_i\ (aT)$
(iia')	$G_{13}\ T_{23} = 0\ (aT)$	or	$G_{14}\ T_{24} = 0\ (aT)$
(iib')	$G_{24}\ T_{14} = 0\ (aT)$		$G_{23}\ T_{13} = 0\ (aT)$

In general, if F_1, F_2 and T are functions of the same variables $x_1, x_2, \ldots x_n$, then $F_1 = F_2$ if F_1 and F_2 have the same decimal specifications and $F_1 = F_2\ (aT)$, (<u>F_1 is equal to F_2 add T</u>), if the decimal specification of F_1 contains some (or all or none) of the numbers in the decimal specification of T in addition to <u>all</u> of the numbers in the decimal specification of F_2.

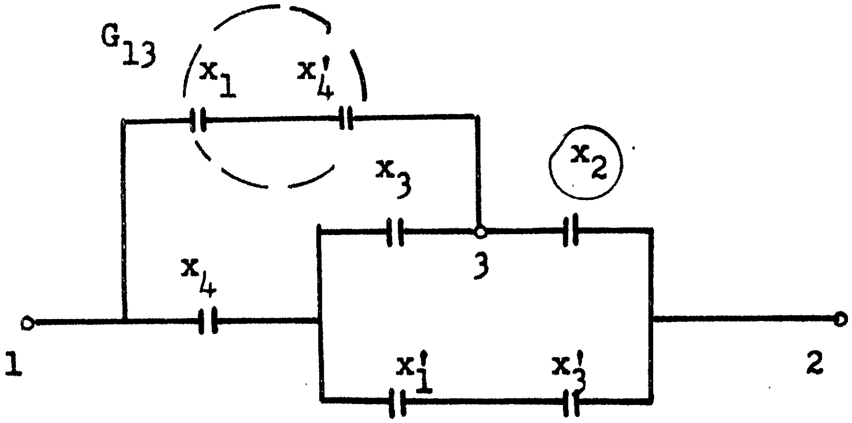
The transmissions T_{13}, T_{34}, etc. which appear in equations (i') and (ii') do not actually have to be total transmissions. The equations are still valid if T_{34} represents the sum of the transmissions of only those paths which do not touch either of nodes 1 or 2 and T_{1j} (T_{2k}) represents the sum of the transmissions of only those paths which do not touch node 2 (1). The proof of this statement is given in Appendix B.

In the networks of Figs. 2.4-1 and 2.4-2, the additional paths were formed without introducing any duplicate contacts. In general, it is not always possible to avoid duplicate contacts. The following section discusses the addition of duplicate contacts.

It is possible for one of the external terminals of the circuit (node 1 or 2) to be an end terminal of a desired path (node 3 or 4). The preceeding discussion still applies in such a situation, but one of the equations (ii') does not apply since only one G_{ij} is to be added. See Fig. 2.4-3.



(a) Network for Transmission $x_4(x_2x_3 + x_1'x_3')$ with Designators which Occur in $x_1x_2x_4'$ Circled



(b) Network Having Transmission $x_4x_3x_2 + x_4x_3'x_1' + x_1x_2x_4'$ Formed by Adding $G_{13} = x_1x_4'$ to the Network of (a)

Figure 2.4-3

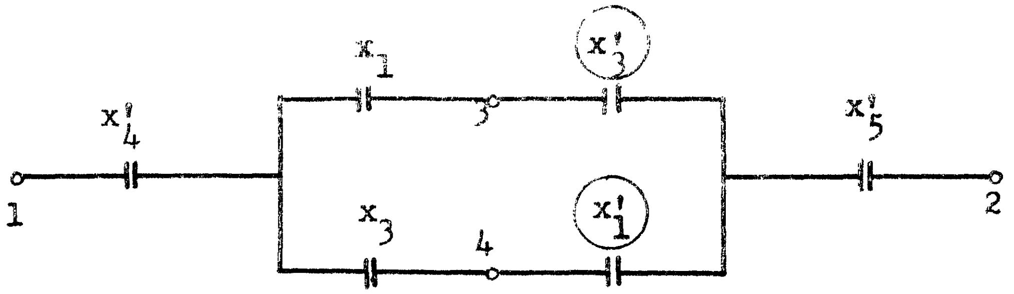
Design of a Contact Network Having Transmission $T = \Sigma(3,7,8,10,14,15) = x_4x_3x_2 + x_4x_3'x_1' + x_4'x_2x_1$

2.5 The Addition of Duplicate Contacts

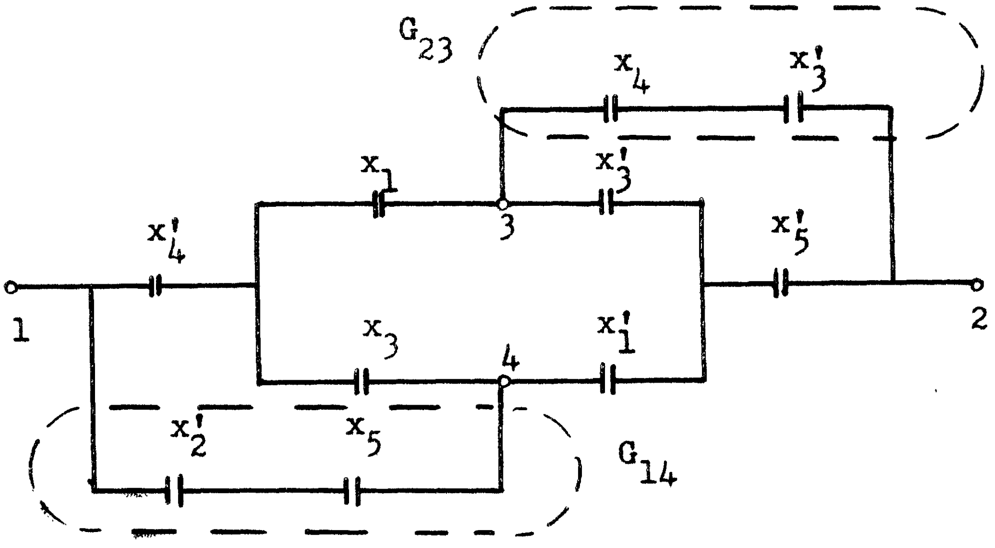
As will be shown in Section 2.6, it is always possible to add any desired path to a network. However, it may be necessary to add duplicate contacts in order to insure that equations (i') and (ii') are satisfied. For example, in Fig. 2.5-1a, $T_{34} = x_1x_3 + x_1'x_3'$ and the transmission of the path to be added is $x_5x_4x_3'x_2'x_1'$. There is in parallel with the desired path a path through which transmission is not desired. In order to prevent the formation of an unwanted sneak path, an x_1' or x_3' contact must be included in either G_{14} or G_{23}. (Fig. 2.5-1b.) This contact blocks transmission through the parallel path x_1x_3.

It is not always possible to block unwanted sneak paths by adding duplicate contacts. However, an unwanted sneak path can always either be blocked or converted into an allowable sneak path by adding duplicate contacts. This statement is proved in Section 2.6. Figure 2.5-2 shows a network in which a duplicate contact is used to form a desirable sneak path.

A path having transmission $x_5'x_4'x_3x_2'$ is to be added to the network of Fig. 2.5-2a. This can be done by choosing $G_{13} = x_2'$ and $G_{24} = x_5'$ as in Fig. 2.5-2b. However, the resulting network contains a sneak path having transmission $x_3x_2'x_1 = \sum(5,13,21,29)$. This is a non-permissible path since



(a) Network for Transmission $x'_4x'_5(x_1x'_3 + x'_1x_3)$ with Designators which Occur in $x_5x_4x'_3x'_2x'_1$ circled



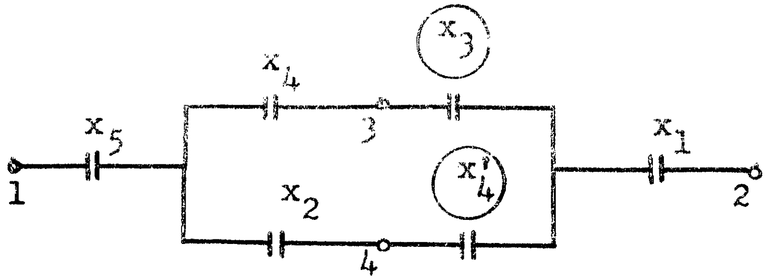
(b) Network Having Transmission $x'_4x'_5(x_1x'_3 + x_3x'_1) + x_5x_4x'_3x'_2x'_1$ Formed by Adding $G_{14} = x'_2x_5$ and $G_{23} = x'_3x_4$ to the Network of (a)

Figure 2.5-1

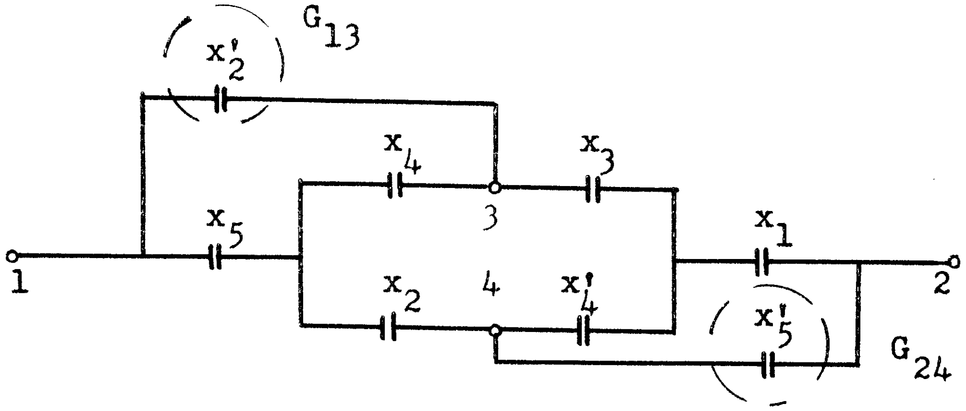
Design of a Contact Network Having Transmission

$T = \Sigma(1,3,4,6,24) = x'_5x'_4x'_3x_1 + x'_5x'_4x_3x'_1 + x_5x_4x'_3x'_2x'_1$

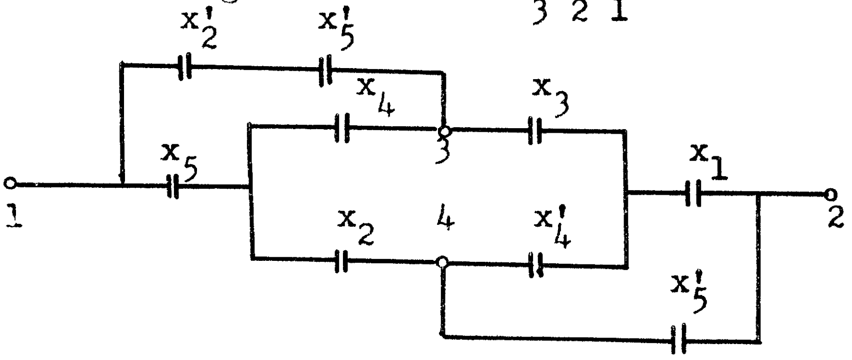
Using one Duplicate Contact



(a) Network for Transmission T = $x_5 x_1 (x_4 x_3 + x_4' x_2)$ with the Designators which Occur in $x_5' x_4' x_3 x_2'$ circled



(b) Network of (a) with G_{13} and G_{24} added to Form a Path Having Transmission $x_5' x_4' x_3 x_2'$. A Sneak Path Having Transmission $x_3 x_2' x_1$ is Present



(c) Network of (b) with Duplicate x_5' Contact Added to Change Sneak Path Transmission to $x_5' x_3 x_2' x_1$

Figure 2.5-2

Design of a Contact Network Having Transmission

T = $\Sigma(4,5,13,19,23,29,31) = x_5 x_4 x_3 x_1 + x_5 x_4' x_2 x_1 + x_5' x_4' x_3 x_2' + x_5' x_3 x_2' x_1$

transmission is not allowed for the condition represented by the (21) term in the decimal specification. By adding a non-essential x_5' contact as in Fig. 2.5-2c, the sneak path transmission is changed to $x_5'x_3x_2'x_1 = \Sigma(5,13)$ which is permissible and, moreover, eliminates the necessity of adding another path to provide transmission for the circuit condition represented by (13).

2.6 A Formal Method for Determining the G_{ij}

Equations (i') and (ii') specify the conditions which the G_{ij} must satisfy. In order to add a desired path to a network, it is necessary to determine somehow those G_{ij} which actually do satisfy (i') and (ii'). Moreover, what is actually wanted is those G_{ij} which while satisfying (i') and (ii') involve the fewest literals and therefore require the fewest contacts in the network. It is possible to devise a formal, "turn the crank", procedure for determining the required G_{ij} from T_{34}, T_{14}, T_{23}, T_{24}, M_i and T.

For a given T_{34}, M_i, and T, it is necessary to determine a transmission $G_{13}\,G_{24}$ (or $G_{14}\,G_{23}$) such that $\underline{G_{13}\,G_{24}\,T_{34} = M_i\ (aT)}$. Thus, $G_{13}\,G_{24}$ is an implicit function of M_i, T_{34} and T. The explicit dependence of $G_{13}\,G_{24}$ on M_i, T_{34}, and T will be determined next. Since a similar relation holds for other sets of transmissions, the discussion will be carried out in terms of general transmissions P, Q, t_1 and T, where P corresponds to $G_{13}\,G_{24}$, Q to T_{34}, t_1 to M_i and T to T. It is assumed that Q is given as a sum of products of literals (not necessarily a standard or minimum sum), $Q = \sum_{i=1}^{s} q_i$; and that t_1 is one of the terms in a similar sum expression, $\sum_{i=1}^{r} t_i$, for T. The transmissions Q, t_1, and T are specified, and the transmission P which contains the fewest variables of all transmissions satisfying $\underline{PQ = t_1\ (aT)}$ is to be found.

This dependence of P on Q, t_1 and T will be signified by writing $P = t_1/Q$ (aT).

The q_i are divided into three classes:

(1) Those q_i for which $q_i \circ t_1 = t_1$

(2) Those q_i for which $q_i \cdot t_1 \neq t_1$ and $q_i \circ t_1 \neq Q$

(3) Those q_i for which $q_i \circ t_1 = 0$

If there are no q_i in class (1) then t_1/Q (aT) is not defined. This corresponds roughly to the situation in ordinary arithmetic where $14 \div 0$ is undefined. If there are no q_i in class (1), it is not possible to find a P such that $PQ = t_1$ (aT). This will never happen when Q corresponds to a T_{34}. In this case, there will always be some q_i which is the transmission of the contact series and which was chosen so that $q_i\ M_i = q_i\ t_i = M_i = t_1$.

Except in the case where $P = t_1/Q$ (aT) is not defined, it is always possible to find some P such that $PQ = t_1$ (aT). If P is chosen as t_1, then $Pq_i = t_1\ q_i = t_1$ for all q_i in class (1), $Pq_i = t_1\ q_i = t_1 \cdot z$ for all q_i in class (2) where z is a product of literals since $t_1\ q_i \neq 0$ and $t_1\ q_i \neq t_1$, and $Pq_i = t_1\ q_i = 0$ for all q_i in class (3). Thus, $PQ = t_1\ Q = t_1 + t_1 + \dots + t_1 + t_1 z + 0 = t_1 = t_1$ (aT) as required.

It is not usually necessary to include all of the literals of t_1 in P. In fact, what is desired is the form of P which contains fewest literals. This P is determined as a product of three transmissions p_1, p_2, and p_3 where p_1 is derived from those q_i in class (1), p_2 is derived from those q_i in class (2), etc. In p_1 will be put only those literals of t_1 which are necessary to make $p_1 \, q_i = t_1$ for all q_i in class (1). In p_2 will be put only those literals of t_1 which are necessary so that $p_2 \, q_i$ will equal one only when T equals one ($p_2 \, q_i$ will be included in T) for all q_i in class (2). Those literals of t_1 which are necessary so that either $p_3 \, q_i$ is included in T or $p_3 \, q_i = 0$ for all q_i in class (3) will be put in p_3.

Specifically let $p_1 = p_{11} \cdot p_{12} \cdots p_{1s}$ where $p_{1i} = 1$ if q_i is in class (2) or (3) and p_{1i} equals t_1 with the literals which appear in q_i removed if q_i is in class (1). If $q_i = t_1$, then p_{1i} is set equal to 1. This definition of p_{1i} guanantees that $p_1 \cdot q_i = t_1$ for all q_i in class (1) and that p_1 will contain the fewest possible literals. This can be stated formally:

if $t_1 = x_1 x_2 \cdots x_n$, $q_i = x_{a_1} \, x_{a_2} \cdots x_{a_k}$

then $p_{1i} = x_{b_1} \, x_{b_2} \cdots x_{b_{n-k}}$

where $\{b_1, \, b_2 \cdots b_{n-k}\} = \{1, 2, \cdots n\} - \{a_1, \, a_2, \cdots a_k\}$

for those q_i in class (1).

<u>Example 1:</u> $T = x_1 x_2 x_3 + x_1' x_2' x_3'$, $t_1 = x_1 x_2 x_3$

$Q = x_1 + x_2 x_3'$, $q_1 = x_1$, $q_2 = x_2 x_3'$

$q_1 \, t_1 = x_1 \cdot x_1 x_2 x_3 = x_1 x_2 x_3 = t_1$ q_1 in class (1)

$q_2 \, t_1 = x_2 x_3' \cdot x_1 x_2 x_3 = 0$ q_2 in class (3)

$p_{12} = 1$, $p_{11} = \cancel{x_1}x_2x_3 = x_2x_3$

$\{b_1, b_2\} = \{1,2,3\} - \{1\} = \{2,3\}$

Example 2: $T = x_1x_2x_3 + x_1'x_2'x_3'$, $t_1 = x_1x_2x_3$

$Q = x_1x_2x_3 + x_2x_3'$, $q_1 = x_1x_2x_3$, $q_2 = x_2x_3'$

$q_1\ t_1 = x_1x_2x_3 \cdot x_1x_2x_3 = x_1x_2x_3 = t_1$ q_1 in class (1)

$q_2\ t_1 = x_2x_3' \cdot x_1x_2x_3 = 0$ q_2 in class (3)

$p_{11} = 1$, $p_{12} = 1$

Example 3: $T = x_1x_2x_3 + x_1'x_2'x_3'$, $x_1x_2x_3 = t_1$

$Q = x_1x_2 + x_2x_3$, $q_1 = x_1x_2$, $q_2 = x_2x_3$

$q_1\ t_1 = x_1x_2x_3 = t_1$, q_1 in class (1) $p_{11} = \cancel{x_1}\cancel{x_2}x_3 = x_3$

$q_2\ t_1 = x_1x_2x_3 = t_1$, q_2 in class (1) $p_{12} = x_1\cancel{x_2}\cancel{x_3} = x_1$

Example 4: $T = x_1x_2x_3 + x_1'x_2'x_3'$, $t_1 = x_1x_2x_3$

$Q = x_1x_2x_3' + x_1x_2x_4$, $q_1 = x_1x_2x_3'$, $q_2 = x_1x_2x_4$

$q_1\ t_1 = x_1x_2x_3' \cdot x_1x_2x_3 = 0$, q_1 is in class (3)

$q_2\ t_1 = x_1x_2x_4 \cdot x_1x_2x_3 = x_1x_2x_3x_4$, q_2 is in class (2)

There is no q_i in class (1), therefore, t_1/Q (aT) is not defined.

Similarly let $p_2 = p_{21} \cdots p_{2s}$ with $p_{2i} = 1$ if q_i is in class (1) or class (3). If q_i is in class (2), p_{2i} is set equal to (t_1 or t_{j_1} or $\cdots$ or t_{j_u}) with the literals which appear in q_i removed from each t. The symbols $t_{j_1} \cdots t_{j_u}$ represent those t_j for which (1) $q_i\ t_j \neq 0$ and (2) the only literals of t_j which do not occur in q_i are literals which do occur in t_1 (t_j for which $t_j q_i t_1 = q_i\ t_1$). Requirement (2) is necessary so that $Pq_i = p_1\ p_2\ p_3\ q_i$ will equal t_1 for those q_i in class (1). The choice of which of the p_{2i}'s should be used must be made so as to have P contain the fewest possible literals. This definition of p_{2i} insures that Pq_i will be included in T ($Pq_i = q_i\ t_j$) for all q_i in class (2). This can be stated formally by making use of the definition already given for p_1 and setting $p_{2i} = t_1\ q_i/q_i$ (a $t_1\ q_i$) or $t_{j_1}\ q_i/q_i$ (a $t_{j_1}\ q_i$) or ... or $t_{j_u}\ q_i/q_i$ (a $t_{j_u}\ q_i$).

<u>Example 5:</u> $T = x_1x_2x_3 + x_1x_2x_4,\ t_1 = x_1x_2x_3$

$Q = x_3 + x_4,\ q_1 = x_3,\ q_2 = x_4$

$q_1\ t_1 = x_3 \cdot x_1x_2x_3 = x_1x_2x_3 = t_1,\ q_1$ is in class (1)

$\underline{p_{11} = x_1x_2},\ \underline{p_{21} = 1}$

$q_2\ t_1 = x_4 \cdot x_1x_2x_3 = x_1x_2x_3x_4,\ q_2$ is in class (2) $\underline{p_{12} = 1}$

$q_2\ t_2 = x_4 \cdot x_1x_2x_4 = t_2; t_2\ q_2\ t_1 = x_1x_2x_3x_4 = q_2\ t_1$

$p_{22} = t_1\ q_2/q_2$ (a $t_1\ q_2$) or $t_2\ q_2/q_2$ (a $t_2\ q_2$)

$p_{22} = x_1x_2x_3x_4/x_4$ (a $x_1x_2x_3x_4$) $= x_1x_2x_3$ or
$x_1x_2x_4/x_4$ (a $x_1x_2x_4$) $= x_1x_2$. $\underline{p_{22} = x_1x_2}$

The q_i in class (3) must have the form $q_i = x'_{c_1}\ x'_{c_2} \cdots x'_{c_m}\ x_{c_{m+1}} \cdots x_{c_j}$ since $t_1\ q_i = 0$. (Recall that $t_1 = x_1x_2 \cdots x_j$). Thus, it is possible to make $Pq_i = 0$ by setting $p_{3i} = x_{c_1}$ or x_{c_2} or ... or x_{c_m} where $p_3 = p_{31}\ p_{32} \cdots p_{3s}$. However, if there are any t_j for which $q_i\ t_j \neq 0$ it may be more economical in certain cases to have $p_{3i} = (t_{j_1}$ or t_{j_2} ... or $t_{j_u})$ with the literals which occur in q_i removed from each t_j. Again the t_{j_1} ... t_{j_u} represent those t_j for which (1) $q_i\ t_j \neq 0$ and (2) the only literals of t_j which do not occur in q_i are literals which do occur in t_1. In general p_{3i} will equal x_{c_1} or x_{c_2} or ... or x_{c_m} or t_{j_1} or t_{j_2} or ... t_{j_u} with the literals which occur in q_i removed from each t_j. The reasons for the requirements on t_{j_1}, t_{j_2}, etc., and for choosing one of the possible p_{3i} are the same as were given in the discussion of p_{2i}. This definition of p_{3i} insures that Pq_i will equal zero or be included in T for all q_i in class (3).

Example 6: $T = x_1x_2x_3 + x_1x_4 + x_2x_4$, $t_1 = x_1x_2x_3$

$t_2 = x_1x_4$, $t_3 = x_2x_4$

$Q = x_3 + x_4 + x_1x_2'$, $q_1 = x_3$, $q_2 = x_4$, $q_3 = x_1x_2'$

$q_1\ t_1 = x_3 \cdot x_1x_2x_3 = t_1$, q_1 is in class (1) $\underline{p_{11} = x_1x_2,\ p_{21} = p_{31} = 1}$

$q_2\ t_1 = x_4 \cdot x_1x_2x_3$, q_2 is in class (2) $\underline{p_{12} = p_{32} = 1}$

$q_2\ t_2 = x_4 \cdot x_1x_4 = t_2$, $t_1\ q_2\ t_2 = x_1x_2x_3 \cdot x_1x_4 = x_1x_2x_3x_4 = q_2\ t_1$

$q_3\ t_3 = x_4 \cdot x_2x_4 = t_3$, $t_1\ q_2\ t_3 = x_1x_2x_3 \cdot x_2x_4 = x_1x_2x_3x_4 = q_2\ t_1$

$p_{22} = t_1\ q_2/q_2$ (a t_1q_2) or $t_2\ q_2/q_2$ (a t_2q_2) or $t_3\ q_2/q_2$ (a t_3q_2)

$= x_1x_2x_3x_4/x_4$ (a $x_1x_2x_3x_4$) or x_1x_4/x_4 (a x_1x_4) or
x_2x_4/x_4 (a x_2x_4)

$p_{22} = x_1x_2x_3$ or x_1 or x_2 = $\underline{x_1 \text{ or } x_2 = p_{22}}$

$q_3\ t_1 = (x_1x_2')(x_1x_2x_3) = 0$, q_3 is in class (3) $\underline{p_{13} = p_{23} = 1}$

$q_3\ t_2 = (x_1x_2')(x_1x_4) = x_1x_2'x_4,\ t_1\ q_3\ t_2 = x_1x_2x_3 \circ x_1x_2'x_4 =$
$0 \neq q_3\ t_2$

$q_3\ t_3 = (x_1x_2')(x_2x_4) = 0$

$\underline{p_{33} = x_2}$ $\quad p_1 = p_{11}\ p_{12}\ p_{13} = x_1x_2 \circ 1 \circ 1 = x_1x_2$

$p_2 = p_{21}\ p_{22}\ p_{23} = 1 \circ x_1$ or $x_2 \circ 1 = x_1$ or x_2

$p_3 = p_{31}\ p_{32}\ p_{33} = 1 \circ 1 \circ x_2$

$P = p_1\ p_2\ p_3 = x_1x_2 = x_1x_2x_3/(x_3 + x_4 + x_1x_2')(a\ x_1x_2x_3 +$
$x_1x_4 + x_2x_4)$

TABLE 2.6-1 SUMMARY OF NOTATION

(1) $P = t_1/Q$ (aT) if $PQ = t_1$ (aT)

(2) $Q = \sum_{i=1}^{s} q_i$

(3) $T = \sum_{i=1}^{r} t_i, \ t_1 = x_1 x_2 \cdots x_n$

(4) q_i is in class (1) if $q_i \cdot t_1 = t_1$

(5) q_i is in class (2) if $q_i \cdot t_1 \neq t_1$ and $q_i \cdot t_1 \neq 0$

(6) q_i is in class (3) if $q_i \cdot t_1 = 0$

(7) $P = p_1 \ p_2 \ p_3$

(8) $p_1 = p_{11} \ p_{12} \cdots p_{1s}$

(9) $p_2 = p_{21} \ p_{22} \cdots p_{2s}$

(10) $p_3 = p_{31} \ p_{32} \cdots p_{3s}$

TABLE 2.6-2 EXPRESSIONS FOR p_{1i}, p_{2i}, AND p_{3i}

p_{1i} - class (1)

p_{1i} = 1 if q_i is in class (2) or (3)

p_{1i} = t_1 with the literals of q_i removed if q_i is in class (1)

= t_1/q_i (a t_1)

p_{2i} - class (2)

p_{2i} = 1 if q_i is in class (1) or (3)

p_{2i} = $t_1 q_i/q_i$ (a t_1 q_i) or t_{j_1} q_i/q_i (a t_{j_1} q_i) or ...
t_{j_u} q_i/q_i (a t_{j_u} q_i)
if q_i is in class (2) where t_{j_1}, t_{j_2} ... t_{j_u} represent those t_j for which q_i $t_j \neq 0$ and t_j q_i t_1 = q_i t_1

p_{3i} - class (3)

p_{3i} = 1 if q_i is in class (1) or (2)

p_{3i} = x_{c_1} or x_{c_2} or ... x_{c_m} or t_{j_1} q_i/q_i (a t_{j_1} q_1) or ...
t_{j_u} q_i/q_i (a t_{j_u} q_i)
where $q_i = x'_{c_1}\ x'_{c_2} \ldots x'_{c_m}\ x_{c_{m+1}} \ldots x_n$ and t_{j_1} t_{j_2} ... t_{j_u} are defined as for p_{2i}.

2.7 Determination of the G_{ij}

When a path is being added to a network, the transmissions T_{34}, T_{12}, T_{13}, T_{23}, T_{24}, M_i and T are known and it is desired to find transmissions G_{13} and G_{24} or G_{14} and G_{23} which satisfy the equations:

$$\begin{array}{llcl}
(i') & G_{13}\,G_{24}\,T_{34} = M_i\ (aT) & & G_{14}\,G_{23}\,T_{34} = M_i\ (aT) \\
(iia') & G_{13}\,T_{23} = 0\ (aT) & \text{or} & G_{14}\,T_{24} = 0\ (aT) \\
(iib') & G_{24}\,T_{14} = 0\ (aT) & & G_{23}\,T_{13} = 0\ (aT)
\end{array}$$

The first step in doing this is to compute: $(G_{13}\ G_{24})_1 = (G_{23}\ G_{14})_1 = M_i/T_{34}$ (aT).* This will insure that a path having transmission M_i is formed. Then $(G_{13})_2 = M_i/(M_i + T_{23})$(aT), $(G_{14})_2 = M_i/(M_i + T_{24})$(aT), $(G_{24})_2 = M_i/(M_i + T_{14})$(aT), $(G_{23})_2 = M_i/(M_i + T_{13})$(aT) are computed. It might be assumed from equations (ii') that $(G_{13})_2 = 0/T_{23}$ (aT), $(G_{14})_2 = 0/T_{24}$ (aT) etc., but this is not correct. The symbol t_1/Q (aT) = 0/Q (aT) is not properly defined. For all q_i, $t_1 \cdot q_i = 0 \cdot q_i = 0 = t_1$ so that all q_i are in class (1) and $P = p_1$. However, $p_{1i} = 0$ and P = 0 when $t_1 = 0$. If $(G_{13})_2 = M_i/(M_i + T_{23})$(aT), then $(G_{13})_2\ (M_i + T_{23}) = M_i$ (aT); but $(G_{13})_2$ by definition contains only literals of M_i so that $(G_{13})_2\ M_i = M_i$ and $M_i + (G_{13})_2\ T_{23} = M_i$(aT). This means that

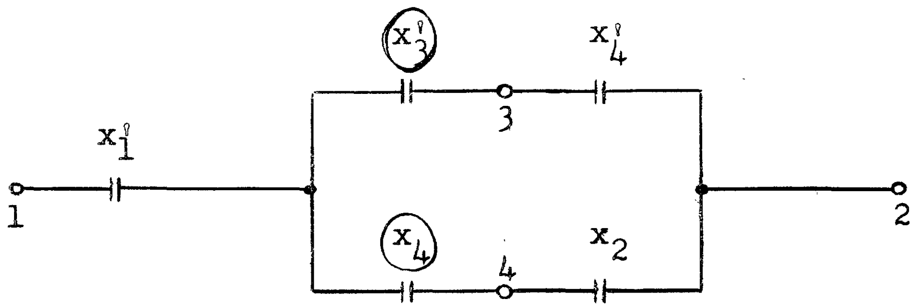
---

* This is done by replacing P by $(G_{13}\ G_{24})_1$, t_i by M_i and Q by T_{34} in the expressions of Section 2.6.

$(G_{13})_2\ T_{23} = 0$ (aT) or M_i (aT), but having $(G_{13})_2\ T_{23} = M_i$ (aT) is equivalent to having $(G_{13})_2\ T_{23} = 0$ (aT) as required. Moreover, if $(G_{13})_2$ is set equal to $M_i/(M_i + T_{23})$(aT), then $(G_{13})_2$ will contain only literals of M_i as required.

There is no guarantee that $(G_{13}\ G_{24})_1 = (G_{13})_2\ (G_{24})_2$ or that $(G_{14}\ G_{23})_1 = (G_{14})_2\ (G_{23})_2$. It may be necessary to add literals to some of these transmissions to make the equations true. For example, if $(G_{13})_2 = x_1$, $(G_{24})_2 = x_2$ and $(G_{13}\ G_{24})_1 = x_1x_2x_3$ then it is necessary to either set G_{13} equal to x_1x_3 and $G_{24} = x_2$ or $G_{13} = x_1$ and $G_{24} = x_2x_3$. Similarly, if $(G_{13})_2 = x_1x_2$, $(G_{24})_2 = x_3$ and $(G_{13}\ G_{24})_1 = x_1x_3$ it is necessary to set $G_{13}\ G_{24} = x_1x_2x_3$.

All four transmissions G_{13}, G_{14}, G_{23}, and G_{24} should usually be computed and the decision as to which should actually be used made so as to add the fewest (if any) duplicate literals. Figure 2.7-1 shows an example where $G_{13}\ G_{24}$ turns out to equal $x_1x_2'x_4$ while $G_{14}\ G_{23}$ equals x_1x_2'. For this network, G_{14} and G_{23} would be the obvious choice.



(a) Network for $T = x_1' (x_3'x_4' + x_4x_2)$ with literals which occur in $M_i = x_4x_3'x_2'x_1$ circled.

$T_{13} = x_1'x_3'$, $T_{14} = x_1'x_4$, $T_{23} = x_4'$, $T_{24} = x_2$, $T_{34} = x_3'x_4 + x_2x_4'$

$T = x_1'x_3'x_4' + x_1'x_2x_4 + x_1x_2'x_3'x_4$, $M_i = x_1x_2'x_3'x_4$

$$(G_{13}G_{24})_1 = (G_{14}G_{23})_1 = M_i/T_{34} \text{ (aT)} = x_1x_2'x_3'x_4/(x_3'x_4 + x_2x_4')\text{(aT)}$$

$$= (x_1x_2')(x_2' \text{ or } x_4) = \underline{x_1x_2'}$$

$$(G_{13})_2 = M_i/M_i + T_{23} \text{ (aT)} = x_1x_2'x_3'x_4/(x_1x_2'x_3'x_4 + x_4')\text{(aT)}$$

$$= 1 \cdot x_4 = \underline{x_4}$$

$$(G_{24})_2 = M_i/M_i + T_{14} \text{ (aT)} = x_1x_2'x_3'x_4/(x_1x_2'x_3'x_4 + x_1'x_4)\text{(aT)} = \underline{x_1}$$

$$(G_{14})_2 = M_i/M_i + T_{24} \text{ (aT)} = x_1x_2'x_3'x_4/(x_1x_2'x_3'x_4 + x_2)\text{(aT)} = \underline{x_2'}$$

$$(G_{23})_2 = M_i/M_i + T_{13} \text{ (aT)} = x_1x_2'x_3'x_4/(x_1x_2'x_3'x_4 + x_1'x_3')\text{(aT)} = \underline{x_1}$$

$$\underline{G_{13}G_{24} = x_1x_2'x_4} \qquad \underline{G_{14}G_{23} = x_1x_2'}$$

(b) Calculations

Figure 2.7-1

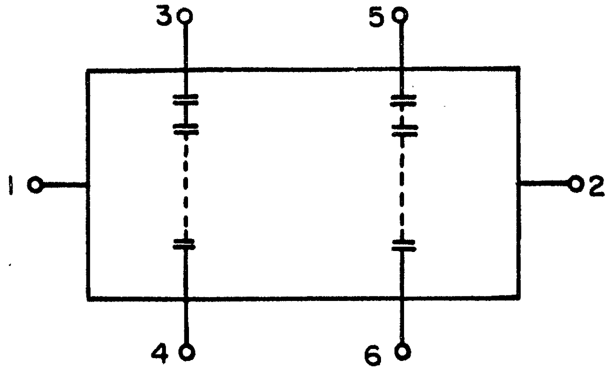
Example of the determination of the G_{ij}

for a network

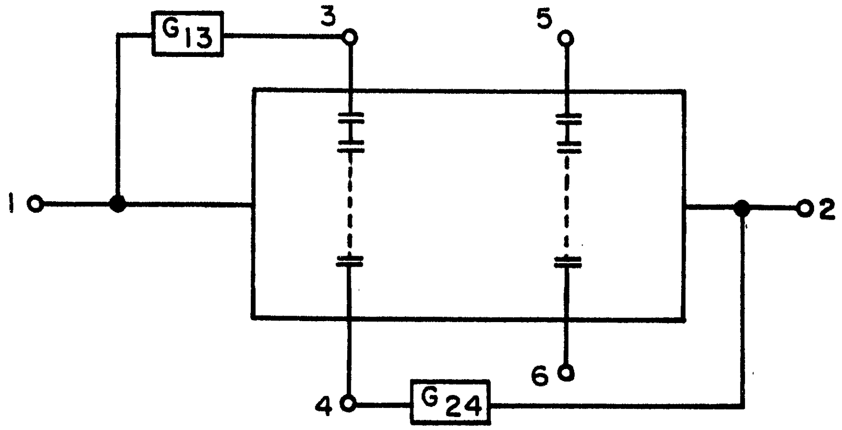
2.8 Multiple Desired Paths

In the preceding sections it has been assumed that the desired contacts only occur so that they form a single desired path. In general, this will not be true. The method of handling single desired paths must be extended to the situation where the desired contacts form an arbitrary subnetwork of the total network. The procedure when several desired paths exist is discussed in this section. The following section presents a method to be used when the desired contacts form loops or stars, etc.

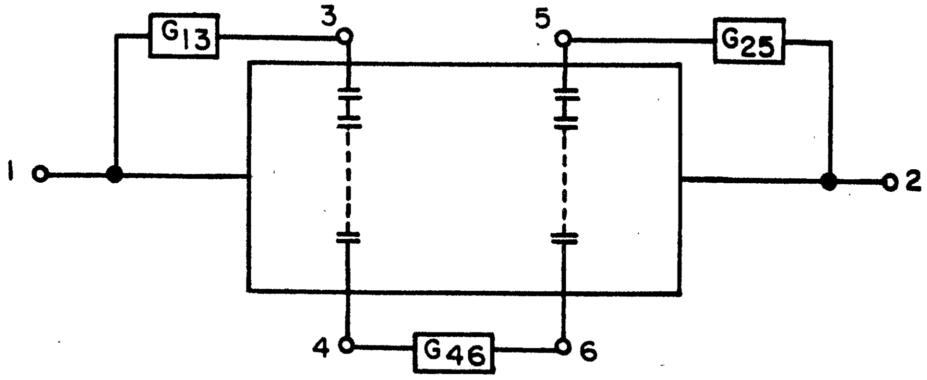
When several desired paths exist as in Fig. 2.8-1a, the method of Section 2.7 is used to form a total path which contains only one of the desired paths (Fig. 2.8-1b). It may then be possible to eliminate some of the contacts in the G_{24} (or G_{13}) network by replacing the G_{24} (or G_{13}) network by two networks G_{46} (G_{45}) and G_{25} (G_{26}) as in Fig. 2.8-1c. This is done to avoid duplicating the contacts of T_{56} in G_{24}. The transmissions G_{46} (G_{45}) and G_{25} (G_{26}) must be chosen so that the total path which includes G_{46} and G_{25} (G_{45} and G_{26}) has a transmission which includes the required transmission M_i, and so that no unwanted sneak paths are formed through G_{46} (G_{45}) or G_{25} (G_{26}). These requirements can be stated formally as:



(a) Network Having Two Desired Paths



(b) Total Path Formed by Including Only T_{34}



(c) Total Path Formed Which Includes Both Desired Paths

Figure 2.8-1

Addition of a Total Path to a Network when Two Desired Paths are Present. (Only Desired Contacts are Shown)

(iii) $G_{46} G_{25} T_{56} T_{14} = M_i$ (aT)

$G_{45} G_{26} T_{56} T_{14} = M_i$ (aT)

(iv) $G_{25} T_{15} = 0$ (aT)

$G_{26} T_{16} = 0$ (aT)

(v) $G_{46} (T_{14} T_{26} + T_{16} T_{24}) = 0$ (aT)

$G_{45} (T_{14} T_{26} + T_{15} T_{24}) = 0$ (aT)

In these equations the T_{ij} are to be calculated for the network which includes G_{13} but not G_{24} since G_{24} is being replaced by G_{46} and G_{25} (or G_{45} and G_{26}).

These equations are correct if T_{ij} represents the sum of the transmissions of all paths between nodes i and j. However, the equations are also correct if instead of all paths between i and j only paths between i and j which do not touch any node whose number occurs as a subscript in the same equation as T_{ij} are considered. For example, T_{56} can be written as the sum of the transmissions of all paths between nodes 5 and 6 which do not touch any of nodes 4,2, or 1 since 4,2, and 1 occur as subscripts in equation (iii). It should be noted that the T_{14} computed for equation (iii) will not be the same as that computed for equation (v) since paths touching node 5 must be considered for equation (v) but not for equation (iii). Since it is always possible to write T_{ij} as the sum of the transmission of <u>all</u> paths between nodes i and j this discussion will not be continued here but will be the subject of Appendix B.

Since equations (iii), (iv) and (v) have the same form as equations (i') and (ii'), the procedure for expressing G_{46}, G_{25}, etc. in terms of T_{56}, T_{14}, etc. is essentially the same as that presented in Section 2.7. Thus:

$(G_{46}\ G_{25}) = (G_{45}\ G_{26}) = M_i/(T_{56}\ T_{14})(aT)$

$(G_{25}) = M_i/(M_i + T_{15})(aT)$, $(G_{26}) = M_i/(M_i + T_{16})(aT)$,

$(G_{46}) = M_i/(M_i + T_{14}\ T_{26} + T_{16}\ T_{24})(aT)$, and

$(G_{45}) = M_i/(M_i + T_{14}\ T_{25} + T_{15}\ T_{24})(aT)$ where

the modifications described in Section 2.7 may be necessary to make $(G_{46}\ G_{25}) = (G_{46})(G_{25})$ or $(G_{45}\ G_{26}) = (G_{45})(G_{26})$.

An example of a network in which two desired paths exist is shown in Fig. 2.8-2a. The symbols t,u,v,w,x,y,z are used instead of $x_8, x_7, x_6, x_5, x_4, x_3, x_2, x_1$ in order to avoid confusion because of numerous subscripts. The first step in forming the required total path is to form a total path which only includes the desired path between nodes 3 and 4. For such a path:

$$(G_{13}\ G_{24}) = (G_{14}\ G_{23}) = M_i/(T_{34})(aT)$$
$$= stu'v'w'xyz'/(x + x'w + t'ys' + t'y'zs')(aT)$$

$$\underline{(G_{13}\ G_{24}) = (G_{14}\ G_{23}) = stu'v'w'yz'}$$

$$(G_{13}) = M_i/(M_i + T_{23})(aT)$$
$$= stu'v'w'xyz'/(stu'v'w'xyz' + v[y't' + zyt' + zs'x + zs'wx'])(aT)$$

$$\underline{(G_{13}) = v' \text{ or } t'z \text{ or } t's' \text{ or } y'z}$$

$$(G_{24}) = M_i/(M_i + T_{14})(aT)$$
$$= stu'v'w'xyz'/(stu'v'w'xyz' + u[w + x't'ys' + x't'y'zs'])(aT)$$

$$\underline{(G_{24}) = u' \text{ or } w'x \text{ or } w't \text{ or } w's}$$

$$(G_{14}) = M_i/(M_i + T_{24})(aT)$$

$$= stu'v'w'xyz'/(stu'v'w'xyz' + v[zs' + zyt'x + zyt'x'w + y't'x + y't'x'w])(aT)$$

$$\underline{(G_{14}) = v' \text{ or } z'y \text{ or } z't \text{ or } st}$$

$$(G_{23}) = M_i/(M_i + T_{13})(aT)$$

$$= stu'v'w'xyz'/(stu'v'w'xyz' + u[x' + wx + ws'yt' + ws'zy't'])(aT)$$

$$\underline{(G_{23}) = u' \text{ or } w'x}$$

In Fig. 2.8-2b the choices $G_{13} = v'$ and $G_{24} = stu'w'yz'$ have been made. It is reasonable to place in G_{24} all literals which can go in either G_{13} or G_{24} since this will allow more freedom in replacing G_{24} by G_{46} and G_{25} (or G_{45} and G_{26}). The other possibility of G_{14} and G_{23} should also be tried, but it will not be shown here. Next the expressions for G_{46}, G_{25}, etc. are calculated.

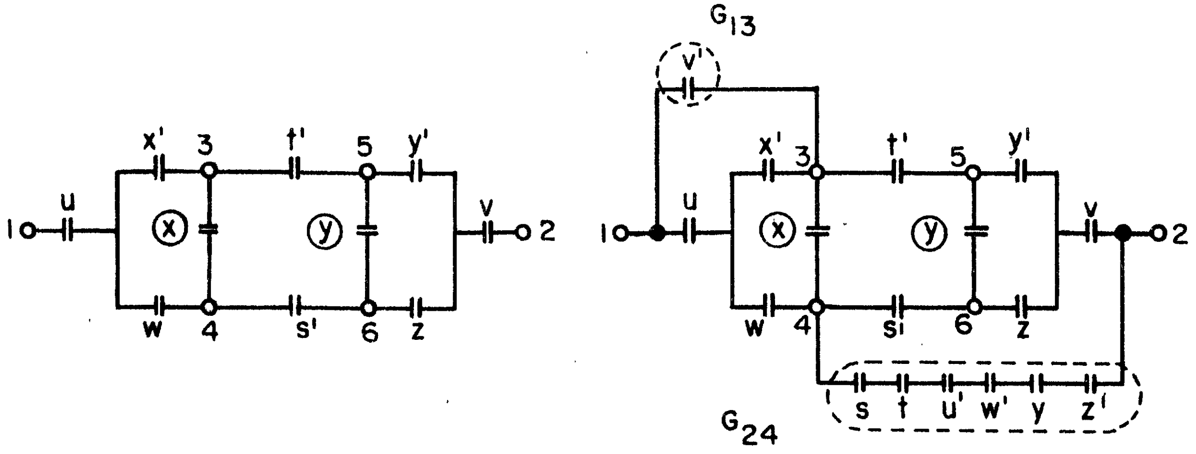
$$(G_{46}\ G_{25}) = (G_{45}\ G_{26}) = M_i/(T_{56}\ T_{14})(aT)$$

$$= stu'v'w'xyz'/(y + y'z)(wu + wx'v + xv')(aT)$$

$$\underline{(G_{46}\ G_{25}) = (G_{45}\ G_{26}) = stu'w'z'}$$

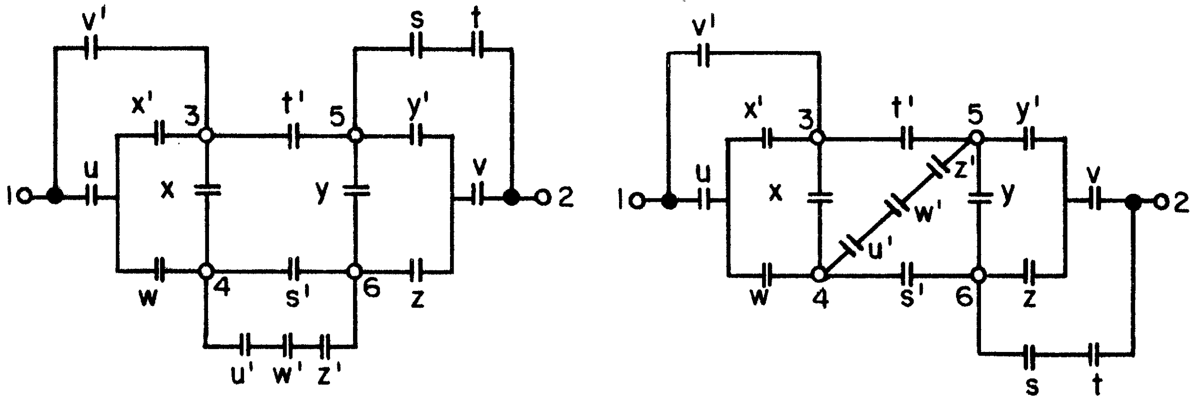
$$(G_{25}) = M_i/(M_i + T_{15})(aT)$$

$$= stu'v'w'xyz'/(stu'v'w'xyz' + t'\ [v' + x'u + xwu] + s'\ [y + y'z][xv' + wu + wx'v'])(aT)$$



(a) Network to Which a Path Having Transmission stu'v'w'xyz' is to be Added

(b) Network of (a) with stu'v'w'xyz' Path Including Only 34 Desired Path



(c) Network of (b) with G_{24} Replaced by G_{25} =st and G_{46} = u'w'z'

(d) Network of (b) with G_{24} Replaced by G_{26} = st and G_{45} = u'w'z'

Figure 2.8-2

Example of the Addition of a Total Path to a Network when Two Desired Paths are Present. (Only Desired Contacts are Shown)

$(G_{25}) = st$

$$(G_{26}) = M_i/(M_i + T_{16})(aT)$$
$$= stu'v'w'xyz'/(stu'v'w'xyz' + s'[wu + wx'v + xv'] + [y + y'z][t'][v' + x'u + xwu])(aT)$$

$(G_{26}) = st$

$$(G_{46}) = M_i/(M_i + T_{14}\ T_{26} + T_{16}\ T_{24})(aT)$$
$$= stu'v'w'xyz'/(stu'v'w'xyz' + vz\ [uw + v'x + v'x'w] + vy't'z\ [v' + x'u][x + x'w])(aT)$$

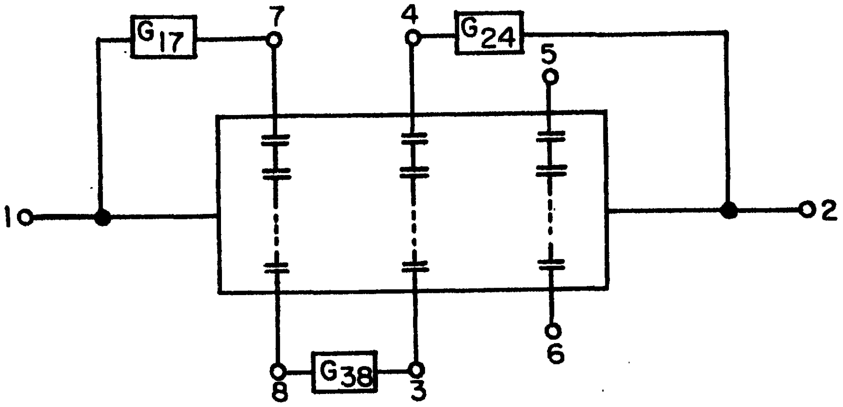
$(G_{46}) = v'$ or z'

$$(G_{45}) = M_i/(M_i + T_{14}\ T_{25} + T_{15}\ T_{24})(aT)$$
$$= stu'v'w'xyz'/(stu'v'w'xyz' + [uv(wy' + wz + s't'x'z)])(aT)$$

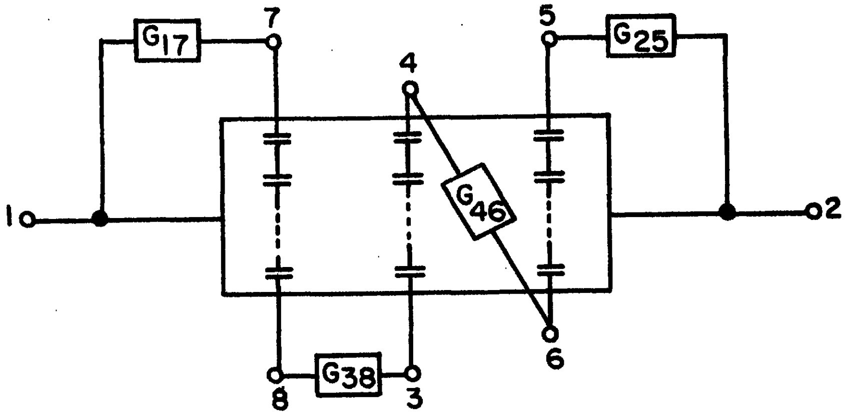
$(G_{45}) = u'$ or v' or $w's$ or $w't$ or $w'x$ or $w'z'$ or yz'

The networks which result from the choices G_{25} = st and G_{46} = u'w'z' or G_{26} = st and G_{45} = u'w'z' are shown in Figs. 2.8-2c and 2.8-2d respectively.

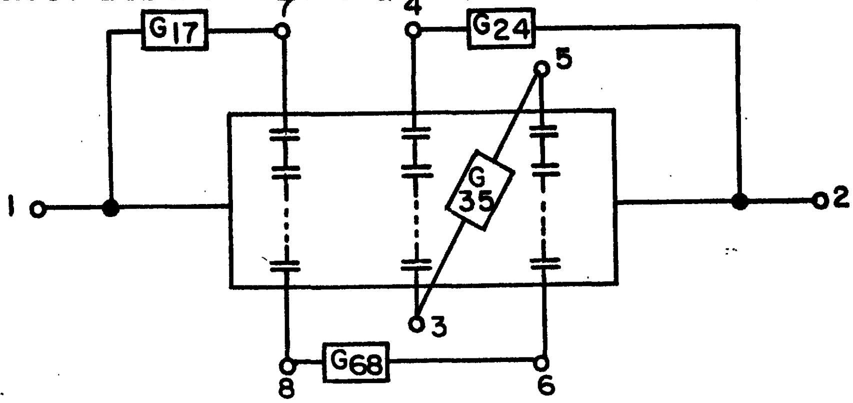
If more than two desired paths are present, the procedure just described is used to form a total path involving two desired paths. The form of the resulting network is shown in Fig. 2.8-3a. It may now be possible to replace G_{24} by G_{25} and G_{46} as in Fig. 2.8-3b, thereby including all three desired paths in the total path. The method for doing this is exactly



(a) Network Having Three Desired Paths for which the Total Path Includes Two Desired Paths



(b) Network of (a) with G_{24} replaced by G_{25} and G_{46} so that all Three Desired Paths are Included in the Total Path



(c) Network of (a) with G_{38} Replaced by G_{68} and G_{35} so that all Three Desired Paths are Included in the Total Path

Figure 2.8-3

The Addition of a Total Path to a Network when Three Desired Paths are Present. (Only Desired Contacts are Shown) the same as that which was used to replace G_{28} by G_{38} and G_{24}; no special consideration need be given to nodes 3,7,or 8.

Another possibility is the replacement of G_{38} by G_{68} and G_{35} (or G_{58} and G_{36}) as shown in Fig. 2.8-3c. The equations relating G_{35} and G_{68} to the T_{ij} are:

(vi) $G_{35}\ (T_{15}\ T_{23} + T_{13}\ T_{25}) = 0\ (aT)$

(vii) $G_{68}\ (T_{18}\ T_{26} + T_{16}\ T_{28}) = 0\ (aT)$

(viii) $G_{68}\ G_{35}\ [T_{18}\ (T_{36}\ T_{25} + T_{56}\ T_{23}) + T_{16}\ (T_{38}\ T_{25} + T_{58}\ T_{23}) + T_{15}\ (T_{36}\ T_{28} + T_{38}\ T_{26}) + T_{13}\ (T_{58}\ T_{26} + T_{56}\ T_{28})] = M_i\ (aT)$

When solved for G_{35} and G_{68} these become:

(vi') $G_{35} = M_i/(M_i + T_{15}\ T_{23} + T_{13}\ T_{25})(aT)$

(vii') $G_{68} = M_i/(M_i + T_{18}\ T_{26} + T_{16}\ T_{28})(aT)$

(viii) $G_{68}\ G_{35} = M_i/(T_{18}\ [T_{36}\ T_{25} + T_{56}\ T_{23}] + T_{16}\ [T_{38}\ T_{25} + T_{58}\ T_{23}] + T_{15}\ [T_{36}\ T_{28}\ T_{26}] + T_{13}\ [T_{58}\ T_{26} + T_{56}\ T_{28}])(aT)$

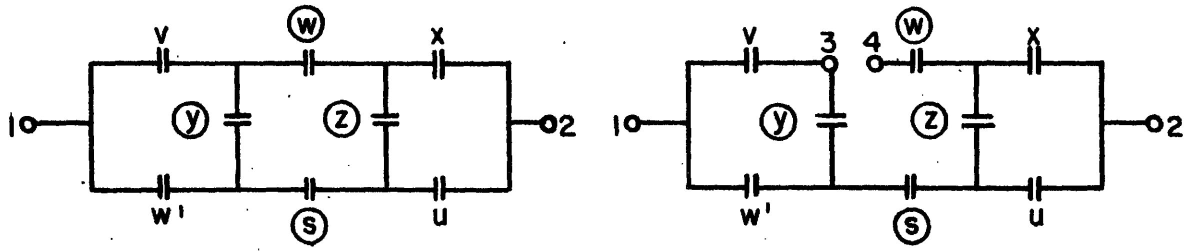
If more than three desired paths are present, the procedure just described is used to form a total path which includes three of the desired paths. The procedure for then modifying the network so that four paths are included in the total path is exactly the same as that which was used to go from a network in which two desired paths are included in the total path to a network in which three desired paths are included in the total path. By repeating this procedure, any number of desired paths can be included in the total path.

2.9 General Desired Contact Subnetworks

In the preceding sections, methods have been presented for forming a total path when separated series connections of desired contacts occur. This section extends these methods to the situations in which the desired contacts occur in any configuration whatsoever. A procedure will be described for modifying a network so that the desired contacts only occur in separate desired paths. Once this has been done, the total path can be formed by the methods of Sections 2.7 and 2.8.

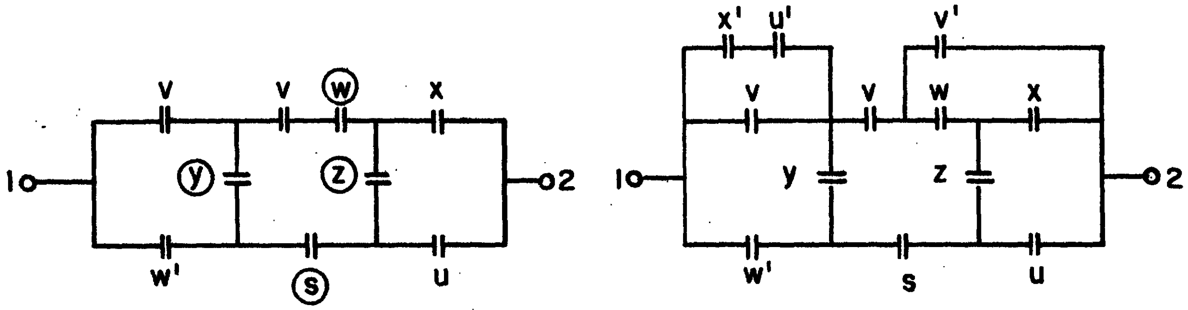
First, the situation where there are loops (closed paths) of contacts as in Fig. 2.9-1a will be considered. A procedure will be developed for adding contacts to the original network so that the over-all transmission is not changed, and there are no longer any desired contact loops. (Fig. 2.9-1c).

The problem to be considered here is that of determining just which contacts can be added to a network without changing the transmission between the external terminals. Fig. 2.9-2 shows the general form of a network in which a wire has been cut open and a contact network of transmission h placed between the ends so formed. (Nodes 3 and 4 in the figure.) In order for the transmission between nodes 1 and 2 to remain unchanged after the insertion of the h-network, it is necessary that $T_{12} = T_{12}$ (h=1) since T_{12} (h=1) is the transmission of the network before insertion of h. By writing T_{12} as a function



(a) Network to which a Path Having Transmission su'v'wx'yz is to be Added

(b) Network of (a) with Desired-Contact Loop Broken



(c) Network of (a) with v Contact Added to Break Desired-Contact Loop

(d) Network of (a) with su'v'wx'yz Path Added

Figure 2.9-1

Example of the Addition of a Total Path to a Network when the Desired Contacts Form a Loop

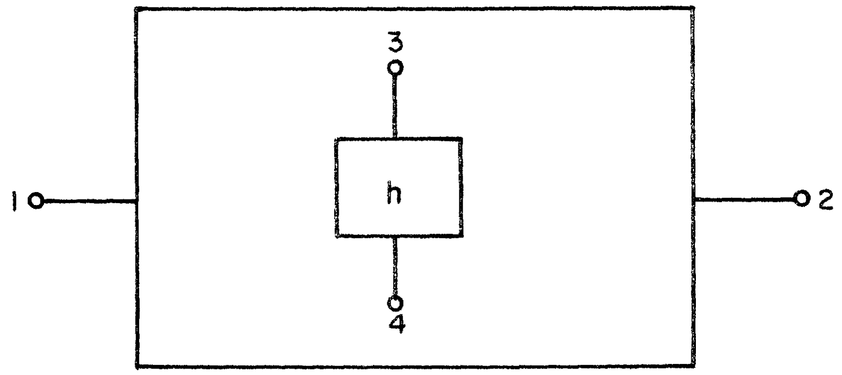


Figure 2.9-2

General Form of Network with Subnetwork of Transmission h Added of t_{12}, t_{13}, etc., where t_{ij} represents the sum of the transmissions of all paths between nodes i and j which do not touch any other numbered node, it is possible to determine the relation between h and the t_{ij}.

Table 2.9-1 is a table of combinations for T_{12} and T_{12} (h=1). From this table it is clear that the only requirement on h is that h equal one when $t_{12}=0$, $t_{34}=0$ and either $t_{13}=1$, $t_{24}=1$, $t_{23}=0$ and $t_{14}=0$ or $t_{13}=0$, $t_{24}=0$, $t_{23}=1$ and $t_{14}=1$. Algebraically, this means that h must include K $(h \supset K)$ where K is defined as $t'_{12}\, t'_{34}\, (t_{12}\, t_{24}\, t'_{23}\, t'_{14} + t'_{13}\, t'_{24}\, t_{23}\, t_{14})$.

In general, K can be expressed as a product of factors and h can be set equal to any single factor or the product of any number of factors. To minimize the number of contacts required, h will usually be set equal to a single factor.

t_{12}	t_{13}	t_{23}	t_{14}	t_{24}	t_{34}	T_{12}	T_{12} (h=1)
1	-	-	-	-	-	1	1
-	1	1	-	-	-	1	1
-	-	-	1	1	-	1	1
0	1	0	0	1	0	h	1
0	0	1	1	0	0	h	1

$$T_{12} = t_{12} + t_{13}\, t_{23} + t_{14}\, t_{24} + (t_{13}\, t_{24} + t_{14}\, t_{23})\, (t_{34} + h)$$

$$T_{12}\ (h=1) = t_{12} + t_{13}\, t_{23} + t_{14}\, t_{24} + t_{13}\, t_{24} + t_{14}\, t_{23}$$

Note: A dash (-) in the table means that the values of T_{12} and T_{12} (h=1) are independent of whether there is a zero or a one in that position in the table.

Table 2.9-1 Table of combinations for T_{12} and T_{12} (h=1)

For the network of Fig. 2.9-1b the calculation of K is as follows:

$$t_{12} = w's\ (u+zx) \qquad t_{34} = wsyz$$

$$t_{13} = v+w'y \qquad t_{14} = 0$$

$$t_{23} = sy\ (u+xz) \qquad t_{24} = w\ (x+uz)$$

$$t'_{12}\ t'_{13} = s'+w\ (y'+z') + w'u'\ (z'+x')$$

$$t_{13}\ t_{24}\ t'_{23}\ t'_{14} = vw\ (x+uz)\ (y'+x'+u'[x'+z'])$$

$$t'_{13}\ t'_{24}\ t_{23}\ t_{14} = 0$$

$$K = vw\ (x+uz)\ (x'+y'+u'z')$$

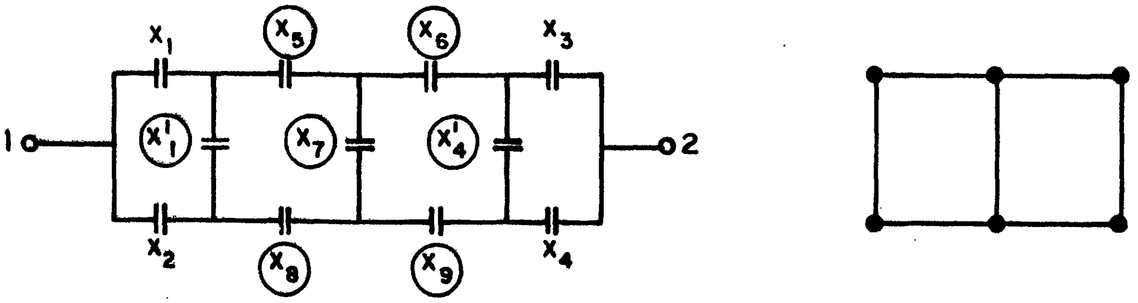
The most economical choice for h is v since w is a trivial possibility and the other factors involve several variables. It is not necessarily true that the most economical choice for h will lead to a network having fewest contacts; however, in this case the choice of v does lead to the best network as shown in Fig. 2.9-1d. The network of Fig. 2.9-1d is obtained from that of Fig. 2.9-1c by means of the procedure of Section 2.7. In general, it may be necessary to try several possibilities for h in order to determine which leads to the most economical network.

The transmission of the total paths which pass through the w-contact in the network of Fig. 2.9-1a is w v(x+uz). From this it is clear that placing a network having transmission w or v or x+uz in series with the w contact will leave the over-all transmission unchanged. It is not directly evident that the insertion of a network with transmission s'+y'+u'z' will also not change the over-all transmission. In this case the transmission of the total paths through w is changed, but the over-all transmission is not. For each state of the network for which the inserted network interrupts transmission through the w-contact there will be transmission through some other total path. For example, when all make contacts are closed (s=u=v=w=x=y=z=1), the network having transmission s'+y'+u'z' would block transmission through the w-contact, but there would still be transmission through the vysu path. This is an example of a situation where it might be expedient to modify a network so that the total paths do not correspond directly to terms of the minimum sum.

Before calculating h it is necessary to decide at what points in the network h-subnetworks are to be placed. When only a single desired contact loop exists it is obvious that h must be placed somewhere in this loop. It may be necessary to calculate h for several locations in the desired contact loop to determine which location requires the fewest contact to be added. When more than one loop exists the possible locations for the h-subnetworks are not so obvious.

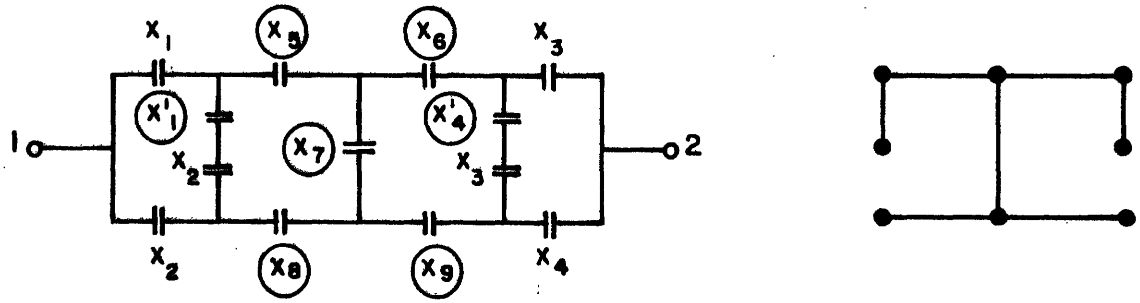
It is convenient to introduce the idea of the <u>graph of a network</u> in order to discuss the problem of locating the h-subnetworks when several desired contact loops exist. In the graph of a network, each contact of the network is represented by a line, called a <u>branch</u>, and the contact terminals are represented by large dots, called <u>nodes</u>, which are placed at the ends of the branches. If several contact terminals are connected together in the network, they will be represented by a single node in the graph. Fig. 2.9-3b shows the graph of the desired contact subnetwork of the network of Fig. 2.9-3a.

Placing an h-subnetwork in a desired contact loop corresponds to breaking the corresponding loop in the graph of the desired contact subnetwork. (See Figs. 2.9-3c and 2.9-3d.) The problem of where to place the h-subnetworks can thus be studied by considering the graph of the desired contact subnetwork. It has been proven [G1, Chapter 1, Section 3; KG 1 Chapter 4, Section 2, Theorem 13] that the number of places where loops must be broken to obtain a graph with no loops is $L = b - n_t + p$ where b is the number of branches, n_t is the number of nodes, and p is the number of unconnected subgraphs of the graph. A graph is called <u>connected</u> if between each pair of its nodes there exists a path. Conversely, two graphs are called <u>unconnected</u> if no path exists from any node of one graph to any node of the other graph. A graph which contains no loops is usually called a "tree". This theorem of graph theory shows that there is a definite minimum to the number of h-subnetworks which must be added to break all desired contact loops.



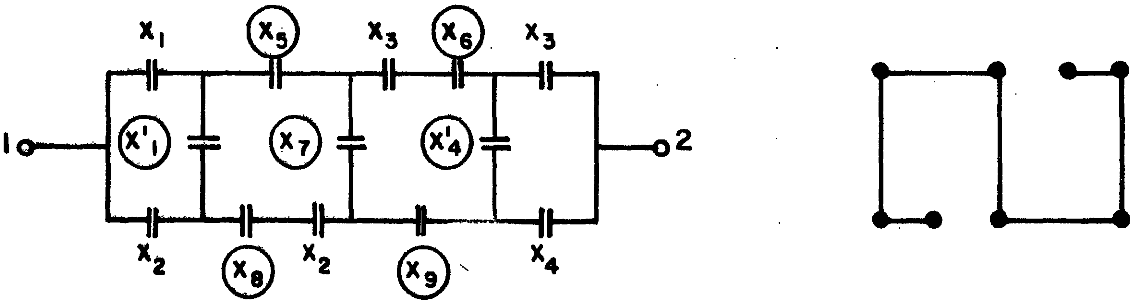
(a) A Contact Network to which a Path Having Transmission $x_9x_8x_7x_6x_5x_4'x_3'x_2'x_1'$ is to be Added

(b) Graph of Desired-Contact Subnetwork of (a)



(c) Network of (a) with x_3 and x_2 Added to Break Desired-Contact Loops

(d) Graph of Desired-Contact Subnetwork of (c)



(e) Network of (a) with x_2 and x_3 Contacts Added so that Only One Desired-Contact Path Remains

(f) Graph of Desired-Contact Subnetwork of (e)

Figure 2.9-3

Example of the Modification of a Network Containing Two Desired-Contact Loops so that Only One Desired-Contact Path Remains

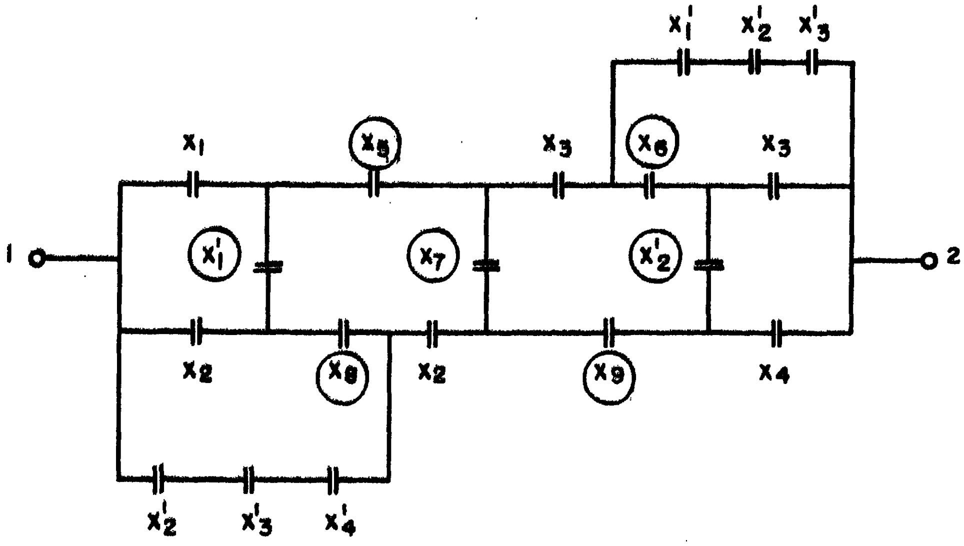


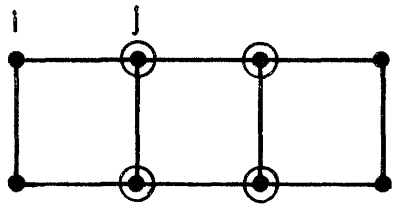
Figure 2.9-4

Final Network for the Example of Fig. 2.9-3

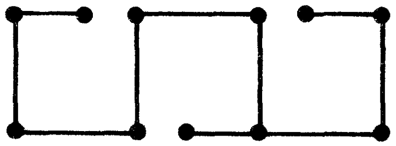
It is not true that all locations of the h-subnetworks are equally appropriate. There is a single desired contact path in the network of Fig. 2.9-3e and therefore, all desired contacts can be included in the total path being formed. For the network of Fig. 2.9-3c this is not true, and either the network must be further modified or the formation of the total path must be carried out without including all of the desired contacts.

It is possible to show that there is at least one arrangement of the h-subnetworks which modifies the network so that only desired contact paths remain. Moreover, the number of h-subnetworks required to do this can be specified (this is not the same number as that required to just break the loops - see Fig. 2.9-5).

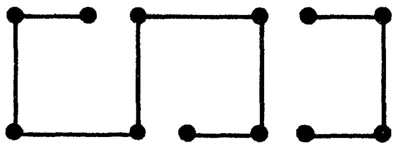
It is first necessary to define the <u>degree</u> of a node as the number of branches which touch the node. For example, in Fig. 2.9-5a node i has degree 2 and node j has degree 3. If a connected graph has 2q nodes of odd degree then the fewest paths into which the graph can be broken is q. Furthermore, each end node of a path will be a node of odd degree. Theorem 3 in Chapter 2 of [KG 1] states that any connected graph contains an even number of nodes of odd degree and the preceeding statement regarding the number of paths is proved as Theorem 4 of the same chapter. For each connected graph, the number of



(a) Graph Containing Three Loops



(b) Graph of (a) with Loops Broken - Three Breaks are Required



(c) Graph of (a) with Loops Broken so that Only Paths Remain - Four Breaks are Required

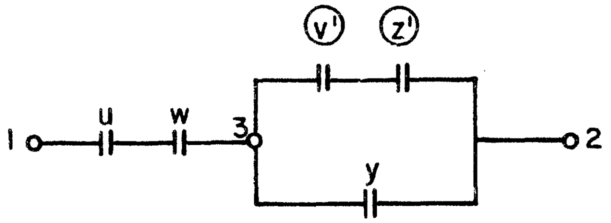
Figure 2.9-5

An Example of the Breaking of Branches in a Graph so that Only Paths Remain places in which branches must be broken is $L' = b-n_t + q$ if there exists nodes of uneven degree and $L = b-n_t + 1$ otherwise. This follows directly from the fact that a graph having q separate paths has exactly q disconnected parts. For the graph of Fig. 2.9-5a there are 2q=4 nodes of odd degree (circled in the Figure), b=10, n_t=8 and $L' = 10-8+2 = 4$ breaks are required as in Fig. 2.9-5c. To summarize, if the graph corresponding to a given desired contact subnetwork has b branches, n_t nodes and 2q nodes of odd degree; then $L' = b-n_t + q$ h-subnetworks must be added so that q desired paths are formed. The determination of the appropriate h-subnetworks is carried out as described above once the location is determined. It should be noted that not all of the h-subnetworks are used to break loops - some are used to separate paths. However, the discussion of the determination of h-subnetworks was completely general and applies directly in such cases.

It is not claimed that there do not exist networks for which it is not desirable to have all the desired contacts occur in paths. This must be decided on an individual basis. In general, it is advisable to break loops, since otherwise none of the contacts in the loop can be used in forming the total path. If neither loops nor separate paths exist (for example, when there are "stars") it is still possible to insert h-subnetworks to form separate paths and the formula for the number of h-subnetworks required ($L' = b-n_t + q$) still holds.

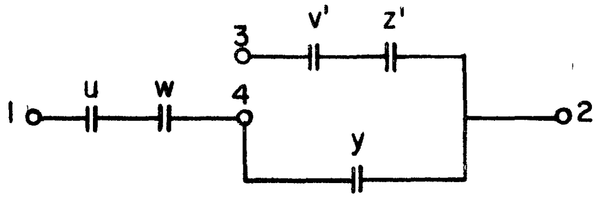
It is sometimes economical to insert an h-subnetwork even when only a single desired contact path exists. For example, in the network of Fig. 2.9-6, a single desired contact path exists and three contacts must be added to form the required total path. By inserting an h-subnetwork (a y' contact) the required total path can be formed with only two additional contacts. In this case it is more economical to block the "sneak" path through the y contact by means of an added y' contact than it is to convert the "sneak" path to a permitted path by adding v' and z' contacts. Whenever several extra contacts must be added so that a "sneak" path has a permitted transmission, the possibility of adding an h-subnetwork to block the sneak path should be considered.

The discussion of how to add a total path of any specified transmission to a given contact network is now complete. The following section will consider the question of how to decide in which order to form the paths.



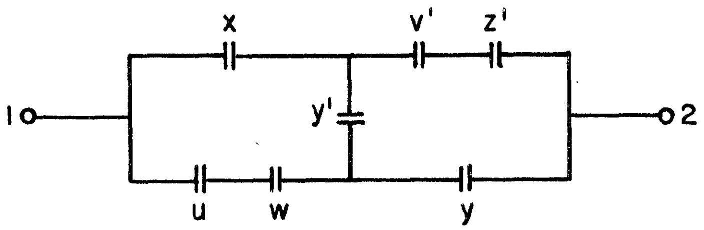
(a) First Step in Designing a Network to Have Transmission

$T = uwv'z' + uwy + xv'z'$; $t_{23} = y + v'z'$; $G_{13} = xv'z'$



(b) Insertion of h-Subnetwork

$t_{12} = t_{34} = t_{13} = 0,\ t_{14} = uw,\ t_{23} = v'z',\ t_{24} = y$
$K = 1\cdot 1\ (0 + 1\cdot y'\cdot v'z'\cdot uw\ ) = uv'wy'z'$



(:) Final Network

Figure 2.9-6

Example of the Use of an h-Subnetwork when a Single Desired-Contact Path Exists

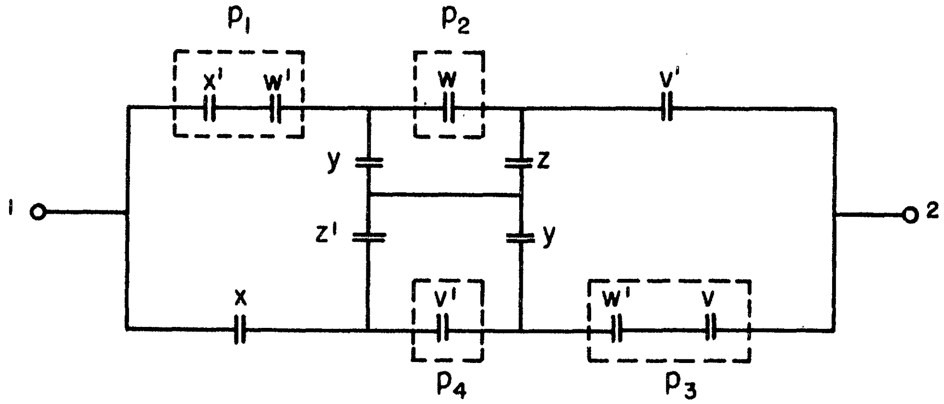
2.10 The Order of Forming Paths

In order to design a contact network to have a specified transmission it is necessary to: (1) write the transmission as a sum of product terms, (2) decide upon the order in which terms will be used, (3) form a network whose transmission equals that of the first two terms, (4) add contacts to the network so that a path having transmission equal to the next term is formed, and (5) repeat step 4 until a path is formed for each term of the transmission. With the exception of (2), (selecting the order in which to use the terms) each of these procedures has been already discussed. The problem of deciding in which order to use the terms of the transmission will be considered in this section.

In general, a given transmission can be realized by many different contact networks. The possibility exists that the order in which the design is carried out may limit the form of the final network. If this is true, the choice of the order is very critical because one order might lead to a more economical network than could be obtained by any other order. Fortunately, it is possible to determine just when the order does limit the form of the network being designed and also what type of limitation is imposed.

A network in which each path contains at least one contact which is not in any other path will be called a <u>distinguished network</u>. An example of a distinguished network is shown in Fig. 2.10-1 in which the contacts which occur in only one path are shown in boxes. The order of design does not restrict the form of a distinguished network. This will be proved by assuming the existence of a counter-example and then showing that the counter-example is not possible. If the statement underlined above is not true, there must exist some distinguished network for which at least one order of design is not possible. Specifically, let the transmissions of the paths through the network be $p_1, p_2, \cdots, p_r$ and let the impossible order of formation be $p_1, p_2, \cdots, p_r$. (The p_i may or may not be terms of a minimum sum and it is assumed that the terms have been numbered to correspond to the impossible order.) In order to disprove the counter-example, a method for determining how to design the network in the "impossible" order will be described. First, contacts will be removed from the network so that the path with transmission p_r is removed and a network with transmission $T = p_1 + p_2 + \cdots + p_{r-1}$ remains. (Fig. 2.10-2a shows the network which results when the p_4 path is removed from the network of Fig. 2.10-1). This procedure is repeated for $p_{r-1}, p_{r-2}, \cdots, p_3$ until a network for only p_2 and p_1 remains.

By carrying out this procedure backwards, it is possible to design the given network in the order $p_1, p_2, \cdots p_r$. To obtain the network for $p_1 + p_2 + \cdots + p_j$ from that for $p_1 + p_2 + \cdots + p_{j-1}$ it is merely necessary to compare the two

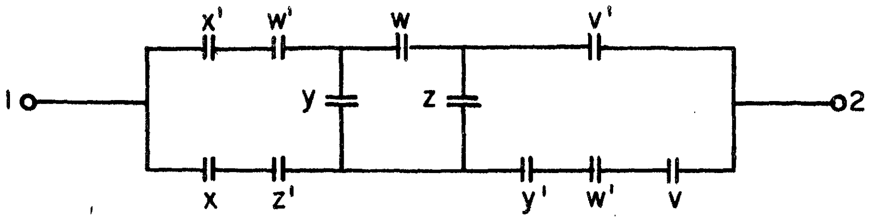


$$T = v'w'x'yz + v'wxyz' + vw'xy'z' + v'xy'z$$

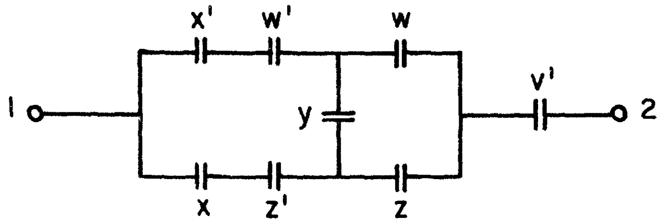
$$\quad p_1 \qquad p_2 \qquad p_3 \qquad p_4$$

Figure 2.10-1

An Example of a Distinguished Network with the Contacts which are Only in One Path Shown in Boxes



(a) Network of Fig. 2.10 with p_4 Path Removed



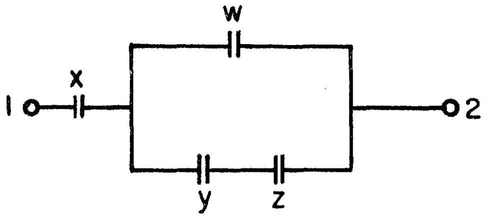
(b) Network of Fig. 2.10-1 with p_3 and p_4 Paths Removed

Figure 2.10-2

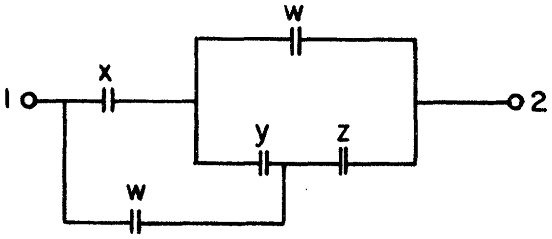
The Network of Fig. 2.10-1 with the p_4 or p_4 and p_3 Paths Removed corresponding networks obtained in the decomposition process described and make the necessary additions to the p_{j-1} network to form the p_j network. This demonstrates the existence of a procedure for forming the network in the given order and thus disproves the counter-example.

A proof has been given for the statement that the form of a distinguished network is not restricted by the order of design. Since the form of the final network is not known when a design is started, this result is only useful if the form of the final network can be related to its transmission. A transmission in which each term contains at least one literal which does not occur in any other term will be called a distinguished transmission. A distinguished transmission can only be realized by means of a distinguished network. Corresponding to each literal which only occurs in one term of a transmission there must be a contact which occurs in only one path of the network realization of that transmission. Since a distinguished transmission has in each term a literal which only occurs in that term, any network realization of a distinguished transmission must have in each path a contact which only occurs in that path. Therefore any network which has a distinguished transmission must be a distinguished network. The converse statement that any distinguished network must have a distinguished transmission is not true. This can easily be seen by considering the network of Fig. 2.10-1. The p_4-term of the transmission does not contain any literal which does not occur in some other term of the transmission. Therefore this is not a distinguished transmission. The network is a distinguished network since the v' contact in the p_4 path is not shared with the v' contact in the p_1 and p_2 paths.

An example of the design of a non-distinguished network is shown in Fig. 2.10-3. In the network of Fig. 2.10-3b each contact is in two paths. It is not possible to remove paths from the network in an arbitrary order since the removal of any contact will remove not one but two paths. Fig. 2.10-3 illustrates the characteristic feature of the design of a non-distinguished network: In the process of forming a path, a useful "sneak" path is also formed. This can never happen with a distinguished transmission since each path requires at least one contact which would not be in the network if the path were not present. For a non-distinguished transmission, the possibility of a useful "sneak" path being formed depends on the order in which paths are added. For the transmission of Fig. 2.10-1, it is possible that p_4 may occur as a "sneak" path, but only if p_1 and p_3 are already in the network so that all contacts required for p_4 will be present. Fig. 2.10-4 shows a network which has the same transmission as the network of Fig. 2.10-1, but which is not a distinguished network.



(a) Step 1 - A Network of Transmission x(w + yz)



(b) Step 2 - The Addition of a w-Contact to Form a wz Path. A "sneak" Path of Transmission wy is also Formed

Figure 2.10-3

An Example of the Design of a Non-Distinguished Network Having Transmission wx + wy + wz + xyz

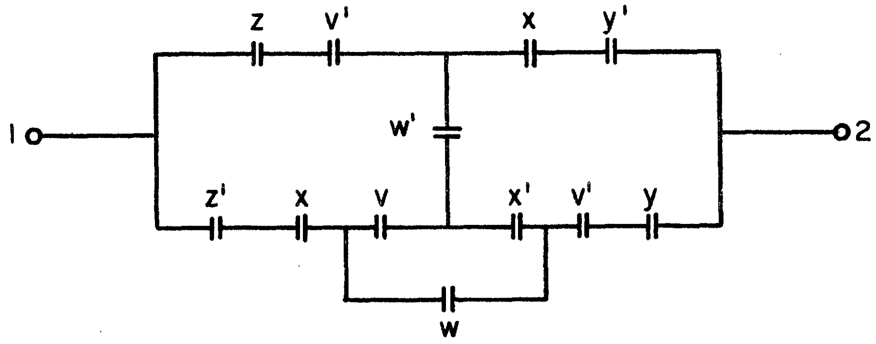


Figure 2.10-4

A Non-Distinguished Network Realization for the Transmission T = v'w'x'yz + v'wxyz' + vw'xy'z' + v'xy'z of Fig. 2.10-1

Even though the p_4 path in the network of Fig. 2.10-4 uses only contacts which occur in other paths, the total number of contacts is the same as in the distinguished network of Fig. 2.10-1. This illustrates the fact that a non-distinguished network is not necessarily more economical than a distinguished network for the same transmission.

To summarize, it has been shown that:

(1) the order of design does not restrict the final form of a distinguished network,

(2) a distinguished transmission must be realized with a distinguished network and a non-distinguished transmission may or may not lead to a distinguished network.

(3) the characteristic of non-distinguished networks is the presence of useful "sneak" paths, and

(4) a non-distinguished network is not necessarily more economical than the equivalent distinguished network.

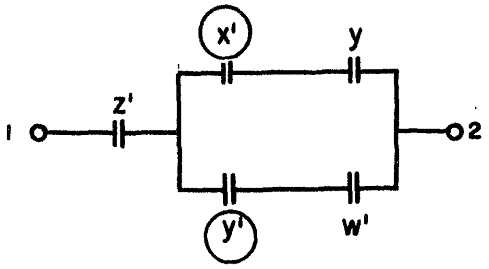
For a non-distinguished transmission it would be possible in principle to try each of the orders of design which could possibly lead to useful "sneak" paths. While this might be reasonable for networks in which economy is extremely important, in general it is not practical because of the large number of trials necessary. In the following discussion it will be assumed that the loss of generality involved in not considering all orders which might lead to useful "sneak" paths can be tolerated and therefore an arbitrary order of design can be followed.

In the preceding discussion, it has been assumed that each path will be formed by adding the fewest possible contacts to the existing network. It is not necessarily true that this procedure will lead to a network having the minimum number of contacts. By using more than the fewest contacts required to form a path, it may be possible to form subsequent paths so that the final network contains fewer contacts than if the minimum number of contacts had been used to add each path. Since there is no way of knowing beforehand when more than the minimum number of contacts should be used or just which "extra" contacts should be inserted, this is not a practical procedure.

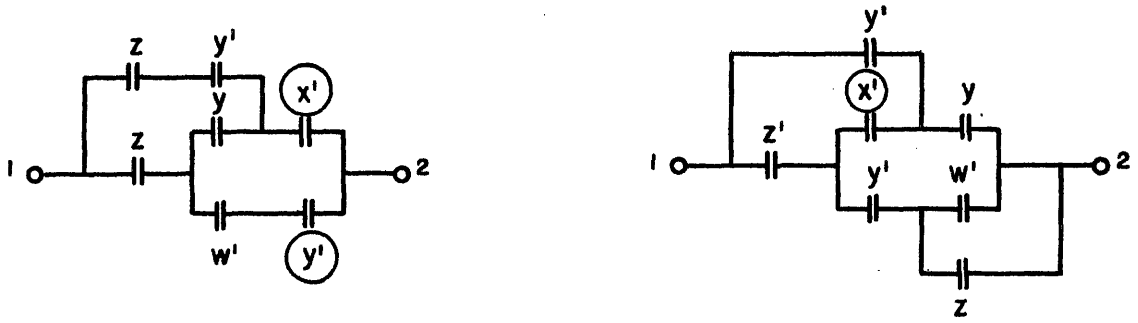
There is always at least one order in which a network having a minimum number of contacts can be obtained by using the fewest contacts at each step. Actually, if using an "extra" contact would lead to economics as described, some other order must exist in which the same network would be designed without using any "extra" contacts. This would involve adding the path with the extra contact at a later stage of the design. These considerations seem to imply that the number of contacts may depend on the order of design if no "extra" contacts are used. However, experience in using this method indicates that generally several different networks involving the same number of contacts may exist for a transmission. Thus for any selected order, there probably is an economical network which does not require the use of "extra" contacts in the design.

A more serious consideration arises from the fact that there are frequently several ways in which the same number of contacts can be used to add a specified path. For example, as shown in Figs. 2.10-5b and 5c, there are two ways to add an x'y'z path to the network of Fig. 2.10-5a. At this stage of the design there is no way to tell which of these ways will lead to a more economical final network. In this case, both procedures have been tried and the final networks (which both use eight contacts) are shown in Figs. 2.10-5d and 5e.

In general, it is not practical to try all the methods which require the same number of contacts, for adding a given path. It is possible that each of these methods would again lead to a situation where a given path could be added in several ways and that a large number of networks would have to be designed. A practical procedure is to postpone adding any paths for which there are several equally economical possibilities and to instead only form paths for which there is only one choice. For the networks of Fig. 2.10-5 this would mean adding the wy'z path before the x'y'z path.

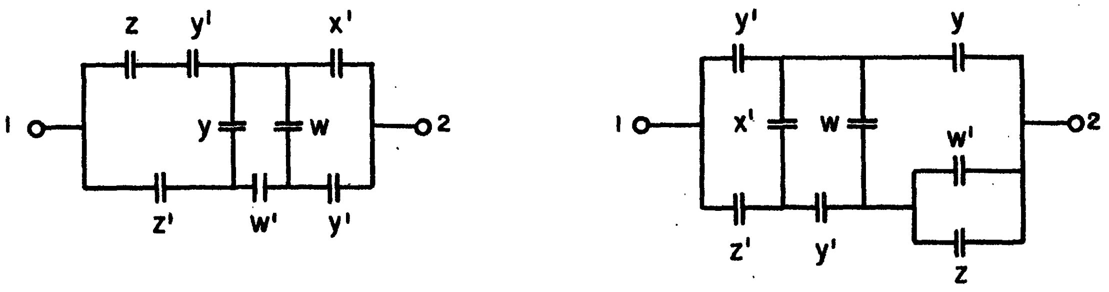


(a) Network for Transmission w'y'z' + x'yz' to which a Path of Transmission x'y'z is to be Added



(b) Network with x'y'z Path Added so that Only the Circled x' Contact is Used

(c) Network with x'y'z Path Added so that both the Circled x' and y' Contacts are Used



(d) Network of (b) with wy'z Path Added

(e) Network of (c) with xy'z Path Added

Figure 2.10-5

An Example of a Network Design in which a Given Path can be Added by Means of Two Methods - Both of which Require the Same Number of Contacts

The question of which order should be used in forming the paths of a network was considered in this section. It was concluded that in practice only one restriction should be placed on the order: An attempt should be made to arrange the order so that in forming a path it is never necessary to choose between two or more methods which require the same number of contacts. The networks designed with only this restriction on the order may not contain the fewest possible contacts. This is because it was decided to add the fewest possible contacts at each step and to ignore the possibility that certain orders might lead to the formation of useful "sneak" paths. These decisions were made solely on the basis of practicality.

These effect of these arbitrary decisions can be avoided by carrying out the design using several orders and then choosing the most economical network. The additional work involved would probably be justified only when economy is extremely important. However, even if several orders of design are used, it is still not possible to guarantee that the resulting network will contain a minimum number of contacts. This is because it was arbitrarily assumed that the transmission of each path through the network was equal to a term of the minimum sum. This will be discussed further in the Summary and Conclusions Section.

SUMMARY AND CONCLUSIONS

In the first part of this thesis a method was presented for writing any transmission as a minimum sum. This method is similar to that of Quine; however, several significant improvements have been made. The notation has been simplified by using the symbols 0, 1 and - instead of primed and unprimed variables. While it is not completely new in itself, this notation is especially appropriate for the arrangement of terms used in determining the prime implicants. Listing the terms in a column which is partitioned so as to place terms containing the same number of 1's in the same partition reduces materially the labor involved in determining the prime implicants. Such a list retains some of the advantage of the arrangement of squares in the Karnaugh Chart without requiring a geometrical representation of an n-dimensional hypercube. Since the procedure for determining the prime implicants is completely systematic it is capable of being programmed on a digital computer. The arrangement of terms introduced here then results in a considerable saving in both time and storage space over previous methods, making it possible to solve larger problems on a given computer. It should be pointed out that this procedure can be programmed on a decimal machine by using the decimal labels instead of the binary characters introduced in Part 1. It is hoped that one of the outgrowths of this thesis will be a digital computer program for determining prime implicants.

A method was presented for choosing the minimum sum terms from the list of prime implicants by means of a Table of Prime Implicants. This is again similar to a method presented by Quine. However, Quine did not give any systematic procedure for handling cyclic prime implicant tables; that is, tables with more than one cross in each column. In Part 1 of this thesis a procedure is given for obtaining a minimum sum from a cyclic prime implicant table. In general, this procedure requires enumeration of several possible minimum sums. If a transmission has any nontrivial group invariances it may be possible to avoid enumeration or to reduce considerably the amount of enumeration necessary. A method for doing this is given. Since there was no satisfactory method known for determining the group invariances of a transmission, it was necessary to develop a procedure for doing this.* This is presented in Appendix A. A modification of this method for detecting

---

* After this was done it was brought to the author's attention that a similar method had been developed independently by Ashenhurst. This was presented in an unpublished report from the Harvard University Computation Laboratory to the Bell Telephone Laboratories.

totally symmetric transmissions is also presented in Appendix A. This method is more systematic than Caldwell's method [C2] and applies for transmissions of any number of variables.

The process of enumeration used for selecting the terms of the minimum sum from a cyclic prime implicant table is not completely satisfactory since it can be quite lengthy. In seeking a procedure which does not require enumeration, the method involving the group invariances of a transmission was discovered. This method is an improvement over complete enumeration, but still has two shortcomings. There are transmissions which have no nontrivial group invariances but which give rise to cyclic prime implicant tables. For such transmissions it is still necessary to resort to enumeration. Other transmissions which do possess nontrivial group invariances still require enumeration after the invariances have been used to simplify the process of selecting minimum sum terms. More research is necessary to determine some procedure which will not require any enumeration for cyclic prime implicant tables. Perhaps the concept of group invariance can be generalized so as to apply to all transmissions which result in cyclic prime implicant tables.

In the second part of this thesis a method for designing two terminal contact networks is presented. This method is different from any of the methods proposed previously. It consists of successively providing paths between the two external terminals which correspond to the individual minimum sum terms. An algebraic procedure is presented for determining which contacts should be added to the existing network in order to provide the required path. Also there are procedures given for determining where the contacts should be added. By use of this method it is possible for the first time to systematically design complex bridge and nonplanar contact networks.

It is not possible to guarantee that every network designed using this method will contain the minimum possible number of contacts. One reason for this is the fact that it has been arbitrarily assumed that each path through the network will correspond to a term of the minimum sum. This will not always hold true for networks having a minimum number of contacts. The method presented here will sometimes require modification of the network so that the paths no longer correspond to minimum sum terms. Although this modification is a systematic result of this method, it is not sufficiently general to lead to all possible networks having a minimum number of contacts. A procedure for avoiding any restriction on the final network is to carry out the design not only for the minimum sum but also for all other sum forms. This is not in general practical. Perhaps after more experience with these networks it will be possible to determine a method for specifying which form of a transmission will lead to a minimum-contact network.

There is the possibility that the order in which the paths are added to the network may influence the number of contacts in the final network. Again this difficulty can be overcome by trying all orders. Unless economy of contacts is extremely important this does not appear practical. Certain restrictions on the order have been developed; however, it is not yet possible to specify which order will lead to a minimum-contact network. It has been shown that there is a certain amount of freedom in selecting the order to be used. Different orders of formation seem to usually lead to different networks which contain the same number of contacts. A practical procedure for selecting an order of formation is given.

Since there is no known method for determining whether a given network contains the minimum possible number of contacts, it is quite difficult to demonstrate whether minimum networks are being designed. The method presented here should always lead to a minimum network if sufficient enumeration is used. If the absolute minimum number of contacts is not required, a direct application of the method should suffice. This method should lead to the design of networks which could not be designed by any previously known systematic procedure.

The minimum sum which is obtained by the method of Part 1 is required for the contact design procedure given in Part 2. In connection with diode logic circuits the minimum sum leads directly to the minimum two stage circuit. The design procedure given in Part 2 is used for designing relay contact networks. It may also be useful for certain types of direct coupled transistor logic circuits.

APPENDIX A

A METHOD FOR DETERMINING THE GROUP INVARIANCES OF A TRANSMISSION

In Section 1.7 it was pointed out that for some transmissions it is possible to permute the variables, or prime some of the variables, or both permute and prime variables without changing the transmission. The following material presents a method for determining for any given transmission which of these operations (if any) can be carried out without changing the transmission.

The permutation operations will be represented symbolically as follows:

$S_{123 \cdots n}$ T will represent the transmission T with no variables permuted

$S_{213 \cdots n}$ T will represent the transmission T with the x_1 and x_2 variables interchanged, etc.

Thus $S_{1432}\ T\ (x_1, x_2, x_3, x_4) = T\ (x_1, x_4, x_3, x_2)$

The symbolic notation for the priming operation will be as follows:

$N_{0000 \cdots 0}$ T will represent the transmission T with no variables primed

$N_{0110 \cdots 0}$ T will represent the transmission T with the x_2 and x_3 variables primed, etc.

Thus $N_{1010}\ T\ (x_1, x_2, x_3, x_4) = T\ (x_1', x_2, x_3', x_4)$.

The notation for the priming operator can be shortened by replacing the binary subscript on N by its decimal equivalent. Thus N_9T is equivalent to N_{0101} T. The permutation and priming operators can be combined. For example, $S_{2134}\ N_3\ T\ (x_1,x_2,x_3,x_4) = T\ (x_2,x_1,x_3',x_4')$. The symbols $S_i\ N_j$ form a mathematical group [BM1], hence the term group invariance.

The problem considered here is that of determining which N_i and S_j satisfy the relation $N_iS_jT = T$ for a given transmission T. Since there are only a finite number of different N_i and S_j operators it is possible in principle to compute $N_i\ S_j\ T$ for all possible $N_i\ S_j$ and then select those $N_i\ S_j$ for which $N_iS_jT = T$. If T is a function of n variables, there are n! possible S_j operators and 2^n N_i operators so that there are n! 2^n possible combinations of $N_i\ S_j$. Actually, if $N_iS_jT = T$ then $N_i\ T$ must equal $S_j\ T$ [S2] so that it is only necessary to compute all $N_i\ T$ and all $S_j\ T$. For $n = 4$, $n! = 24$ and $2^n = 16$ so that the number of possibilities to be considered is quite large even for functions of only four variables. It is possible to avoid enumerating all $N_i\ T$ and $S_j\ T$ by taking into account certain characteristics of the transmission being considered.

The first step in determining the group invariances of a transmission is the same as that for finding the prime implicants. The binary equivalents of the decimal numbers which specify the transmission are listed as in Fig. A-1a. This list of binary numbers will be called the <u>transmission</u>

	x_1	x_2	x_3	x_4
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1

(a) Transmission Matrix

	x_1	x_2	x_4	x_3
0	0	0	0	0
2	0	0	1	0
1	0	0	0	1
10	1	0	1	0
9	1	0	0	1
11	1	0	1	1

(b) Transmission Matrix with x_3 and x_4 columns interchanged

	x_1	x_2	x_3'	x_4'
3	0	0	1	1
2	0	0	1	0
1	0	0	0	1
10	1	0	1	0
9	1	0	0	1
8	1	0	0	0

(c) Transmission Matrix with entries of the x_3 and x_4 columns primed.

Figure A-1

Transmission matrices showing effect of interchanging or priming variables matrix. When two variables are interchanged, the corresponding columns of the transmission matrix are also interchanged. (Fig. A-1b.) When a variable is primed, the entries in the corresponding column of the transmission matrix are also primed (0 replaced by 1 and 1 replaced by 0, Fig. A-1c).

If a N_i S_j operation leaves a transmission unchanged then the corresponding matrix operations will not change the transmission matrix aside from possibly reordering the rows. In other words, it should be possible to reorder the rows of the modified transmission matrix to regain the original transmission matrix. For example, the matrices of Figs. A-1a and A-1b are identical except for the interchange of the 1 and 2 and the 9 and 10 rows. It is not possible to make the matrix of Fig. A-1c identical with that of Fig. A-1a by reordering rows; therefore the operation of priming the x_3 and x_4 variables does not leave the transmission $T = \Sigma(0,1,2,9,10,11)$ unchanged.

If interchanging two columns of a matrix does not change the matrix aside from rearranging the rows, then the columns which were interchanged must both contain the same number of 1's (and 0's). This must be true since rearranging the rows of a matrix does not change the total number of 1's in each column. Similarly, if priming some columns of a matrix leaves the rows unchanged, either each column must have an equal number of 1's and 0's or else for each primed column which has an unequal number of 0's and 1's there must be a second primed column which has as many 1's as the first primed column has 0's and vice versa. Such pairs of columns must also be interchanged to keep the total number of 1's in each column invariant. For the matrix of Fig. A-2a the only operations which need be considered are either interchanging x_1 and x_2 or x_3 and x_4 or priming and interchanging x_5 and x_6.

For the present it will be assumed that no columns of the matrix have an equal number of 0's and 1's. It is possible to determine all permuting and priming operations which leave such a matrix unchanged by considering only permutation operations on a related matrix. This related matrix, called the standard matrix, if formed by priming all the columns of the original matrix which have more 1's than 0's (the x_6 column in the matrix of Fig. A-2a). Each column of a standard matrix must contain more 0's than 1's (Fig. A-2b). The N_i S_j operations which leave the original matrix unchanged can be determined directly from the operations which leave the corresponding standard matrix unchanged. It is therefore only necessary to consider standard matrices.

Since no columns of a standard matrix have an equal number of 1's and 0's and no columns have more 1's than 0's it is only necessary to consider permuting operations. The

	x_1	x_2	x_3	x_4	x_5	x_6
4	0	0	0	1	0	0
8	0	0	1	0	0	0
5	0	0	0	1	0	1
9	0	0	1	0	0	1
33	1	0	0	0	0	1
7	0	0	0	1	1	1
11	0	0	1	0	1	1
49	1	1	0	0	0	1
30	0	1	1	1	1	0
Number of 0's	7	7	5	5	6	3
Number of 1's	2	2	4	4	3	6

(a) Transmission Matrix

	x_1	x_2	x_3	x_4	x_5	x_6'	Weight
4	0	0	0	1	0	0	1
8	0	0	1	0	0	0	1
32	1	0	0	0	0	0	1
5	0	0	0	1	0	1	2
6	0	0	0	1	1	0	2
9	0	0	1	0	0	1	2
10	0	0	1	0	1	0	2
48	1	1	0	0	0	0	2
31	0	1	1	1	1	1	5
	7	7	5	5	6	6	
	2	2	4	4	3	3	

(b) Standard Matrix for (a) Matrix

x_1	x_2	x_3	x_4	x_5	x_6'
0	0	0	1	0	0
0	0	1	0	0	0
1	0	0	0	0	0
0	0	0	1	0	1
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	0
1	1	0	0	0	0
0	1	1	1	1	1

(c) Second Partitioning of rows for (b) matrix

x_1	x_2	x_3	x_4	x_5	x_6'
0	0	0	1	0	0
0	0	1	0	0	0
1	0	0	0	0	0
0	0	0	1	0	1
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	0
1	1	0	0	0	0
0	1	1	1	1	1

(d) Final Partitioning for (b) matrix

Figure A-2
Partitioning of the standard matrix for
$T = \sum(4,5,7,8,9,11,30,33,49)$

number of 1's in a column (or row) will be called the weight of the column (or row). Only columns or rows which have the same weights can be interchanged. The matrix should be partitioned so that all columns (or rows) in the same partition have the same weight (Fig. A-2b). It is now possible to interchange columns in the same column partition and check whether pairs of rows from the same row partition can then be interchanged to regain the original matrix. This can usually be done by inspection. For example, in Fig. A-2b if columns x_4 and x_3 are interchanged, then interchanging rows 4 and 8, 5 and 9, and 6 and 10 will regain the original matrix.

The process of inspection can be simplified by carrying the partitioning further. In the matrix of Fig. A-2b, row 32 cannot be interchanged with either row 8 or row 4. This is because it is not possible to make row 32 identical with either row 8 or row 4 by interchanging columns x_1 and x_2. Row 32 has weight 1 in these columns while rows 8 and 14 both have weight 0. In general, only rows which have the same weight in each submatrix can be interchanged. Permuting columns of the same partition does not change the weight of the rows in the corresponding submatrices. The matrix can therefore be further partitioned by separating the rows into groups of rows which have the same weight in every column partition (Fig. A-2c). Similar remarks hold for the columns so that it may then be necessary to partition the columns again so that each column in a partition has the same weight in each submatrix (Fig. A-2d). Partitioning the columns may make it necessary to again partition the rows, which in turn may make more column partitioning necessary. This process should be carried out until a matrix results in which each row (column) of each submatrix has the same weight. Inspection is then used to determine which row and column permutations will leave the matrix unchanged. Only permutations among rows or columns in the same partition need be considered.

From the matrix of Fig. A-2d it can be seen that permuting either columns x_3 and x_4 or columns x_5 and x_6' will not change the matrix aside from reordering certain rows. This means that interchanging x_3 and x_4 or priming and interchanging x_5 and x_6 in the original transmission will leave the transmission unchanged. Interchanging x_6' and x_5 means replacing x_5 by x_6' and x_6 by x_5' which is the same as interchanging x_5 and x_6 and then priming both x_5 and x_6. Thus for the transmission of Fig. A-2 $S_{123456}\ T = T$ and $N_{000011}\ S_{123465}\ T = N_3\ S_{123465}\ T = T$.

A procedure has been presented for determining the group invariance of any transmission matrix which does not have an equal number of 1's and 0's in any column. This must now be extended to matrices which do have equal numbers of 0's and 1's in some columns (Fig. A-3a). For such matrices the procedure is to prime appropriate columns so that there are either more 0's than 1's or the same number of 0's and 1's in each column (Fig. A-3a). This matrix is then partitioned as described above and the permutations which leave the matrix unchanged are determined. The matrix of Fig. A-3a is

$$\begin{array}{rcc|cc}
 & x_1 & x_2 & x_3 & x_4 \\
0 & 0 & 0 & 0 & 0 \\ \cline{2-5}
6 & 0 & 1 & 1 & 0 \\
9 & 1 & 0 & 0 & 1 \\ \cline{2-5}
12 & 1 & 1 & 0 & 0 \\
\text{Number of 0's} & 2 & 2 & 3 & 3 \\
\text{Number of 1's} & 2 & 2 & 1 & 1
\end{array}
\qquad
\begin{array}{rcc|cc}
 & x_1' & x_2' & x_3 & x_4 \\
0 & 0 & 0 & 0 & 0 \\ \cline{2-5}
10 & 1 & 0 & 1 & 0 \\
5 & 0 & 1 & 0 & 1 \\ \cline{2-5}
12 & 1 & 1 & 0 & 0 \\
 & 2 & 2 & 3 & 3 \\
 & 2 & 2 & 1 & 1
\end{array}$$

(a) Transmission Matrix

(b) Transmission Matrix with x_1 and x_2 primed

Figure A-3

Transmission matrices for $T = \Sigma(0,6,9,12)$

so partitioned. Interchanging both x_1 and x_2, and x_3 and x_4 leave this matrix unchanged so that $S_{2143}\ T = T$. The possibility of priming different combinations of the columns which have an equal number of 0's and 1's must now be considered. Certain of the possible combinations can be excluded beforehand.

In Fig. A-3a the only possibility which must be considered is that of priming both x_1 and x_2. If only x_1 or x_2 is primed, there will be no row which has all zeros. No permutation of the columns of this matrix (with x_1 or x_2 primed) can produce a row with all zeros. Therefore this matrix cannot possibly be made equal to the original matrix by rearranging rows and columns. Priming both x_1 and x_2 must be considered since the 12-row will be converted into a row with all zeros. The operation of priming x_1 and x_2 is written symbolically as $N_{1100} = N_{12}$. In general, if the matrix has a row consisting of all zeros, only those N_i operations for which i is the number of some row in the matrix need be considered. If the row does not have an all-zero row only those N_i for which i is <u>not</u> the number of some row need be considered. Similarly, if the matrix has a row consisting of all 1's, only those N_i for which there is some row of the matrix which will be converted into an all-one row need be considered. This is equivalent to considering only those N_i for which some row has a number $k = 2^n-1-i$* where n is the number of columns. If the matrix does <u>not</u> have an all-one row, only those N_i for which <u>no</u> row has a number $k = 2^n-1-i$ should be considered.

---

* The number of the row which has all ones is 2^n-1. If N_i operating on some row, k, is to produce the all-one row i must have 1's wherever k has 0's and vice versa. This means that

$$i + k = 2^n-1 \text{ or } k = 2^n-1-i.$$

Each priming operation which is not excluded by these rules is applied to the transmission matrix. The matrices so formed are then partitioned as described previously. Any of these matrices that have the same partitioning as the original matrix are then inspected to see if any row and column permutations will convert them to the original matrix. For the matrix of Fig. A-3a the operation of priming both x_1 and x_2 was not excluded. The matrix which results when these columns are primed is shown in Fig. A-3b. Inspection of this figure shows that interchange of either x_3 and x_4 or x_1' and x_2' will convert the matrix back to the matrix of Fig. A-3a. Therefore, for the transmission of this figure S_{1243} N_{1100} $T = T$ and S_{2134} N_{1100} $T = T$.

There are certain transmissions whose value depends not on which relays are operated but only on how many relays are operated. For example, the transmission of Fig. A-4a equals 1 whenever two or three relays are operated. For such transmissions any permutation of the variables leaves the transmission unchanged. These transmissions are called <u>totally symmetric</u> [S1]. They are usually written in the form, $T = S_{a_1, a_2 \cdots a_m} (x_1, x_2, \cdots x_n)$, where the transmission is to equal 1 only when exactly a_1 or a_2 or $\cdots$ or a_m of the variables $x_1, x_2 \cdots x_n$ are equal to one. The transmission of Fig. A-4a can be written as $S_{2,3} (x_1, x_2, x_3)$. This definition of symmetric transmissions can be generalized by allowing some of the variables $(x_1, x_2, \cdots x_n)$ to be primed. Thus the transmission S_3 (x_1, x_2', x_3) will equal 1 only when $x_1 = x_2' = x_3 = 1$ or $x_1 = x_3 = 1$ and $x_2 = 0$. It is useful to know when a transmission is totally symmetric since special design techniques exist for such functions [C1,KRW1].

$$\begin{array}{cccc} & x_3 & x_2 & x_1 \\ 3 & 0 & 1 & 1 \\ 5 & 1 & 0 & 1 \\ 6 & 1 & 1 & 0 \\ 7 & 1 & 1 & 1 \end{array}$$

Figure A-4

Transmission matrix for

$$T = \sum (3,5,6,7) = S_{2,3} (x_1, x_2, x_3)$$

It is possible to determine whether a transmission is totally symmetric from its matrix. Unless all columns of the standard matrix derived from the transmission matrix have the same weight, the transmission cannot possibly be totally symmetric. If all columns do have equal weights, the rows should be partitioned into groups of rows which all have the same weight. Whether the transmission is totally symmetric can now be determined by inspection. If there is a row of weight k; that is, a row which contains k 1's, then every possible row of weight k must also be included in the matrix. This means that there must be $_nC_k$ rows of weight k where n is the number of columns (variables). If any possible row of weight k was not included then the corresponding k literals could be set equal to 1 without the transmission being equal to 1. This contradicts the definition of a totally symmetric transmission. In Fig. A-5b there are 4 rows of weight 1 and 1 row of weight 4. Since $_4C_1 = 4$ and $_4C_4 = 1$ this transmission is totally symmetric and can be written as $S_{1,4}\ (x_1, x_2', x_3, x_4')$. The number of rows of weight 1 in Fig. A-5d is 2 and since $_4C_1 = 4$ this transmission is <u>not</u> totally symmetric.

---

* $_nC_k$ is the binomial coefficient $\frac{n!}{(n-k)!k!}$

$$\begin{array}{lrccccrcccc} & & x_1 & x_2 & x_3 & x_4 & & x_1 & x_2' & x_3 & x_4' \\ & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ & 4 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 1 & 0 \\ \cline{3-6} & 10 & 1 & 0 & 1 & 0 & 4 & 0 & 1 & 0 & 0 \\ \cline{3-6} & 7 & 0 & 1 & 1 & 1 & 8 & 1 & 0 & 0 & 0 \\ \cline{8-11} & 13 & 1 & 1 & 0 & 1 & 15 & 1 & 1 & 1 & 1 \\ \cline{3-6} \cline{8-11} \text{Number of 0's} & & 3 & 2 & 3 & 2 & & 3 & 3 & 3 & 3 \\ \text{Number of 1's} & & 2 & 3 & 2 & 3 & & 2 & 2 & 2 & 2 \end{array}$$

(a) Transmission Matrix for $T = \Sigma(1,4,7,10,13)$

(b) Standard Matrix for $T = \Sigma(1,4,7,10,13)$ showing that $T = S_{1,4}(x_1, x_2', x_3, x_4')$

$$\begin{array}{lrccccrcccc} & & x_1 & x_2 & x_3 & x_4 & & x_1' & x_2' & x_3 & x_4' \\ & 3 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \cline{8-11} & 5 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ & 10 & 1 & 0 & 1 & 0 & 8 & 1 & 0 & 0 & 0 \\ \cline{8-11} & 12 & 1 & 1 & 0 & 0 & 7 & 0 & 1 & 1 & 1 \\ \cline{3-6} & 13 & 1 & 1 & 0 & 1 & 14 & 1 & 1 & 1 & 0 \\ \cline{3-6} \cline{8-11} \text{Number of 0's} & & 2 & 2 & 3 & 2 & & 3 & 3 & 3 & 3 \\ \text{Number of 1's} & & 3 & 3 & 2 & 3 & & 2 & 2 & 2 & 2 \end{array}$$

(c) Transmission Matrix for $T = \Sigma(3,5,10,12,13)$

(d) Standard Matrix for $T = \Sigma(3,5,10,12,13)$ showing that it is not totally symmetric

Figure A-5

Determination of totally symmetric transmissions

APPENDIX B

JUSTIFICATION FOR SIMPLIFICATION OF TRANSMISSION EXPRESSIONS IN SECTIONS 2.4 AND 2.8

In section 2.4 it was stated that total transmissions for T_{ij} were not required in the equations:

(i') $G_{13}G_{24}T_{34} = M_i(aT)$ $\quad$ $G_{14}G_{23}T_{34} = M_i(aT)$

(iia') $G_{13}T_{23} = O(aT)$ $\quad$ or $\quad$ $G_{14}T_{24} = O(aT)$

(iib') $G_{24}T_{14} = O(aT)$ $\quad$ $G_{23}T_{13} = O(aT)$

This statement will now be proved. T_{34} can be written as $T_{34} = t_{34} + t_{13}t_{14} + t_{23}t_{24} + t_{23}t_{12}t_{14} + t_{13}t_{12}t_{24}$ where t_{ij} represents the sum of the transmissions of all paths between nodes i and j which do not touch any other numbered nodes. In terms of the t_{ij}, equation (i') becomes $G_{13}G_{24}(t_{34} + t_{13}t_{14} + t_{23}t_{24} + t_{23}t_{12}t_{14} + t_{13}t_{12}t_{14}) = M_i(aT)$. Since $t_{12} = O(aT)$, $G_{13}G_{24}(t_{23}t_{14} + t_{13}t_{14})t_{12} = O(aT)$ so that the $t_{23}t_{12}t_{14}$ and $t_{13}t_{12}t_{14}$ terms can be dropped from this expression. Equations (ii') require that $G_{13}t_{23} = O(aT)$ and $G_{24}t_{14} = O(aT)$ so that $G_{24}(G_{13}t_{23})t_{24} = O(aT)$ and $G_{13}(G_{24}t_{14})t_{13} = O(aT)$ and the $t_{13}t_{14}$ and $t_{23}t_{24}$ terms can be dropped from (i'). All that remains is $G_{13}G_{24}t_{34} = M_i(aT)$ as stated in Section 1.4.

T_{23} can be written as $T_{23} = t_{23} + t_{12}t_{13} + t_{24}t_{34} + t_{12}t_{24}t_{34} + t_{24}t_{14}t_{34}$. Equation (iia') becomes $G_{13}(t_{23} + t_{12}t_{13} + t_{24}t_{34} + t_{12}t_{14}t_{34} + t_{24}t_{14}t_{13}) = O(aT)$. Again

$t_{12} = O(aT)$ so that $(t_{13} + t_{14}t_{34})(t_{12}) = O(aT)$ and the $t_{12}t_{13}$ and $t_{12}t_{24}t_{34}$ terms can be dropped from (iia'). The transmission $t_{14}t_{24}$ is included in T_{12}. Since $T_{12} = O(aT)$ then $t_{14}t_{24} = O(aT)$ and therefore $t_{13}(t_{14}t_{24}) = O(aT)$ so that the $t_{24}t_{14}t_{13}$ term can be removed from (iia'). Equation (iia') has been reduced from $G_{13}T_{23} = O(aT)$ to $G_{13}(t_{23} + t_{24}t_{34}) = O(aT)$ or T_{23} has been replaced with the transmission of only those paths between nodes 2 and 3 which do not touch node 1. If the numbers 1 and 2 interchanged throughout this proof, the corresponding statement for equation (iib') is proved.

In Section 2.8 it was stated that the T_{ij} for equations (iii), (iv) and (v) could be set equal to the sum of the transmissions of all paths between nodes i and j which do not touch any node whose number appears as a subscript, in the equations in which T_{ij} appears.
These equations in question are:

(iii) $G_{46}G_{25}T_{56}T_{14} = M_i(aT)$

(iv) $G_{25}T_{15} = O(aT)$

(v) $G_{46}(T_{14}T_{26} + T_{16}T_{24}) = O(aT)$

Let t^{α}_{ij} be the sum of the transmissions of all paths between nodes i and j which do not touch any other nodes whose numbers appear as subscript in equation (iii). Thus t^{α}_{56} is the sum of the transmission of all paths between nodes 5 and 6 which do not touch any of nodes 1, 2, or 4. The symbols t^{β}_{ij} and t^{γ}_{ij} will have the same definition as t^{α}_{ij} except that t^{β}_{ij} will refer to equation (iv) and t^{γ}_{ij} will refer to equation (v). It is possible to write the T_{56} in equation (iii) as:

$T_{56} = t^{\alpha}_{56} + A + B + C$ where:

$A = t^{\alpha}_{15}\ [t^{\alpha}_{16} + t^{\alpha}_{12}(t^{\alpha}_{26} + t^{\alpha}_{24}\ t^{\alpha}_{46})]$,

$B = t^{\alpha}_{25}\ [t^{\alpha}_{26} + t^{\alpha}_{12}(t^{\alpha}_{16} + t^{\alpha}_{14}\ t^{\alpha}_{46})]$,

$C = t^{\alpha}_{45}\ [t^{\alpha}_{46} + t^{\alpha}_{24}(t^{\alpha}_{26} + t^{\alpha}_{12}\ t^{\alpha}_{16})]$.

Equation (iii) requires that:

(iii') $G_{46}G_{25}T_{56}T_{14} = G_{46}G_{25}T_{14}(t^{\alpha}_{56} + A + B + C) = M_1(aT)$

Since T_{12} includes only terms of T, $T_{12} = O(aT)$. It is also true that $t^{\alpha}_{12} = O(aT)$ since t^{α}_{12} is included in T_{12}. Thus all the terms which contain t^{α}_{12} can be dropped from equation (iii') which then simplifies to:

(iii") $G_{46}G_{25}T_{14}\ [t^{\alpha}_{56} + t^{\alpha}_{15}\ t^{\alpha}_{16} + t^{\alpha}_{25}\ t^{\alpha}_{26} +$

$t^{\alpha}_{45}\ t^{\alpha}_{46} + t^{\alpha}_{45}\ t^{\alpha}_{24}\ t^{\alpha}_{26}] = M_1(aT)$

Equation (iv) requires that $G_{46}T_{14}(G_{25}t^{\alpha}_{15}) = O(aT)$ so that this term can be dropped from (iii"). Equation (v) requires that $G_{25}(t^{\alpha}_{25} + t^{\alpha}_{45}\ t^{\alpha}_{24})\ (G_{46}T_{14}t^{\alpha}_{26}) = O(aT)$ so that the $t^{\alpha}_{25}t^{\alpha}_{26}$ and $t^{\alpha}_{45}t^{\alpha}_{24}t^{\alpha}_{26}$ terms can also be dropped from equation (iii"). Since T_{15} includes $t^{\alpha}_{45}\ t^{\alpha}_{14}$, equation (iv) requires that $G_{25}t^{\alpha}_{45}t^{\alpha}_{14} = O(aT)$ so that the $t^{\alpha}_{45}t^{\alpha}_{46}$ term can be dropped from (iii''). The equation (iii'') has thus been reduced to $G_{46}G_{25}T_{14}t^{\alpha}_{56} = M_i(aT)$. A similar expansion of T_{14} in terms of t^{α}_{ij} shows that T_{14} can be replaced by t^{α}_{14}. Equation (iii) is thus shown to be equivalent to $G_{46}G_{25}t^{\alpha}_{14}t^{\alpha}_{56} = M_i(aT)$ as was stated in Section 2.8.

In terms of t^{β}_{ij}, equation (iv) becomes $G_{25}(t^{\beta}_{15} + t^{\beta}_{12}t^{\beta}_{25}) = O(aT)$. For the same reasons that $t^{\alpha}_{12} = O(aT)$, $t^{\beta}_{12} = O(aT)$ so that $t^{\beta}_{12}t^{\beta}_{25}$ can be dropped from this equation which then reduces to $G_{25}t^{\beta}_{15} = O(aT)$ as required.

Equation (v) can be written in terms of t^{γ}_{ij} as:

$$G_{46}[T_{26}(t^{\gamma}_{14} + t^{\gamma}_{12}\,[t^{\gamma}_{24} + t^{\gamma}_{26}t^{\gamma}_{46}] + t^{\gamma}_{16}[t^{\gamma}_{46} + t^{\gamma}_{26}t^{\gamma}_{24}]) + T_{16}\,(t^{\gamma}_{24} + t^{\gamma}_{12}[t^{\gamma}_{14} + t^{\gamma}_{16}t^{\gamma}_{46}] + t^{\gamma}_{26}[t^{\gamma}_{46} + t^{\gamma}_{16}t^{\gamma}_{14}])] = O(aT).$$

Again all t^{γ}_{12} terms can be dropped because of the reason given for dropping t^{γ}_{12} terms from (iii''). Also $T_{26}t^{\gamma}_{16}$ and $T_{16}t^{\gamma}_{26}$ are both included in T_{12} so that $T_{26}t^{\gamma}_{16} = O(aT)$ and $T_{16}t^{\gamma}_{26} = O(aT)$ and all terms containing $T_{26}t^{\gamma}_{16}$ or $T_{16}t^{\gamma}_{26}$ can be dropped. Thus it is possible to reduce equation (v) to $G_{46}[T_{26}t^{\gamma}_{14} + T_{16}t^{\gamma}_{24}] = O(aT)$. Expansion of T_{26} and T_{16} in terms of t^{γ}_{ij} shows that they can be replaced by t^{γ}_{26} and t^{γ}_{16} in equation (v). Therefore equation (v) can be written in terms of only t^{γ}_{26}, t^{γ}_{14}, t^{γ}_{16} and t^{γ}_{24} as stated in Section 2.8.

BIBLIOGRAPHY

[BM1] Birkhoff, G., and MacLane, S., A Survey of Modern Algebra. The MacMillan Company, New York.

[C1] Caldwell, S. H., Switching Circuits and Logical Systems, Teaching Notes for Course 6.567, M.I.T.

[C2] Caldwell, S. H., "The Recognition and Identification of Symmetric Switching Circuits," Trans.A.I.E.E., vol. 73, part I (1954), pp. 142-146.

[G1] Guillemin, E. A., Introductory Circuit Theory. John Wiley and Sons, Inc., 1953.

[H1] Huffman, D. A., "The Synthesis of Sequential Switching Circuits," Journal of the Franklin Institute, vol. 257, no. 3, March, 1954, pp. 161-303.

[HS1] Hohn, F. E., and Schissler, L. R., "Boolean Matrices and the Design of Combinational Switching Circuits," Bell System Technical Journal, vol. 34, pp. 177-202, January, 1955.

[KG1] Konig, Denes, Theorie der Endlichen und Unendlichen Graphen, Chelsea Publishing Company, New York, New York, 1950.

[K1] Karnaugh, M., "The Map Method for Synthesis of Combinational Logic Circuits," Trans.A.I.E.E., vol. 72, part I (1953), pp. 593-598.

[KRW1] Keister, W., Ritchie, A. E., Washburn, S., The Design of Switching Circuits, New York, D. Van Nostrand Company, Inc., 1951.

[M1] Moore, E. F., Unpublished Work.

[MN1] Montgomerie, G. A., "Sketch for an Algebra of Relay and Contactor Circuits," Journal I.E.E., vol. 95, part 3, 1948, pp. 303-312.

[Q1] Quine, W. V., "The Problem of Simplifying Truth Functions," The American Mathematical Monthly, vol. 59, no. 8, pp. 521-531, October, 1952.

[S1] Shannon, C. E., "A Symbolic Analysis of Relay and Switching Circuits," Trans. A.I.E.E., vol. 57 pp. 713-723, 1938.

[S2] Shannon, C. E., "The Synthesis of Two-Terminal Switching Circuits," Bell System Technical Journal, vol. 28, pp. 59-98, 1949.

[SCL1] Staff of the Harvard Computation Laboratory, Synthesis of Electronic Computing and Control Circuits, Cambridge, Mass., 1951, Harvard University Press.

[V1] Veitch, E. E., "A Chart Method for Simplifying Truth Functions," Proc. Association for Computing Machinery, May, 1952.

BIOGRAPHY

Edward Joseph McCluskey, Jr. was born on October 16, 1929 in Brooklyn, New York. He entered Bowdoin College in 1947 and transferred to the Massachusetts Institute of Technology in 1950 under the Bowdoin - M.I.T. Combined Plan. In June 1953 he graduated summa cum laude from Bowdoin College with an A.B. degree in Mathematics and Physics and received the B.S. and M.S. degrees in Electrical Engineering from M.I.T. He is a member of Phi Beta Kappa, Tau Beta Pi, Eta Kappa Nu, and an associate member of Sigma Xi. From September 1952 to January 1953 he was a lecturer in Physics at Newton College of the Sacred Heart. At M.I.T. he was a research assistant from September 1953 to February 1955 and an instructor from February 1955 to September 1955. At present, he is a member of the technical staff at the Bell Telephone Laboratories and a lecturer in Electrical Engineering at the City College of New York. He is married and has two children.