A SYMBOLIC ANALYSIS
OF
RELAY AND SWITCHING CIRCUITS

by

Claude Elwood Shannon
B.S., University of Michigan
1936

Submitted in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE
from the
Massachusetts Institute of Technology
1940

Signature of Author_____
Department of Electrical Engineering, August 10, 1937

Signature of Professor
in Charge of Research_____

Signature of Chairman of Department
Committee on Graduate Students_____

TABLE OF CONTENTS

ACKNOWLEDGMENT

The author is indebted to Professor F. L. Hitchcock, who supervised the thesis, for helpful criticism and advice.

I Introduction: Types of Problems

In the control and protective circuits of complex electrical systems it is frequently necessary to make intricate interconnections of relay contacts and switches. Examples of these circuits occur in automatic telephone exchanges, industrial motor control equipment and in almost any circuits designed to perform complex operations automatically. Two problems that occur in connection with such networks of switches will be treated here. The first, which will be called analysis, is to determine the operating characteristics of a given circuit. It is, of course, always possible to analyze any given circuit by setting up all possible sets of initial conditions (positions of switches and relays) and following through the chain of events so instigated. This method is, however, very tedious and open to frequent error.

The second problem is that of synthesis. Given certain characteristics, it is required to find a circuit incorporating these characteristics. The solution of this type of problem is not unique and it is therefore additionally desirable that the circuit requiring the least number of switch blades and relay

contacts be found. Although a solution can usually be obtained by a "cut and try" method, first satisfying one requirement and then making additions until all are satisfied, the circuit so obtained will seldom be the simplest. This method also has the disadvantages of being long, and the resulting design often contains hidden "sneak circuits."

The method of solution of these problems which will be developed here may be described briefly as follows: Any circuit is represented by a set of equations, the terms of the equations representing the various relays and switches of the circuit. A calculus is developed for manipulating these equations by simple mathematical processes, most of which are similar to ordinary algebraic algorisms. This calculus is shown to be exactly analogous to the Calculus of Propositions used in the symbolic study of logic. For the synthesis problem the desired characteristics are first written as a system of equations, and the equations are then manipulated into the form representing the simplest circuit. The circuit may then be immediately drawn from the equations. By this method it is always possible to find the simplest circuit containing only series and parallel connections,

and for certain types of functions it is possible to find the simplest circuit containing any type of connection. In the analysis problem the equations representing the given circuit are written and may then be interpreted in terms of the operating characteristics of the circuit. It is also possible with the calculus to obtain any number of circuits equivalent to a given circuit.

Phraseology will be borrowed from ordinary network theory for concepts in switching circuits that are roughly analogous to those of impedence networks.

II Series-Parallel Two Terminal Circuits

Fundamental Definitions and Postulates. We shall limit our treatment to circuits containing only relay contacts and switches, and therefore at any given time the circuit between any two terminals must be either open (infinite impedance) or closed (zero impedance). Let us associate a symbol X_{ab} or more simply X, with the terminals a and b. This variable, a function of time, will be called the hinderance of the two terminal circuit a-b. The symbol 0 (zero) will be used to represent the hinderance of a closed circuit, and the symbol 1 (unity) to represent the hinderance of an open circuit. Thus when the circuit a-b is open $X_{ab} = 1$ and when closed $X_{ab} = 0$. Two hinderances X_{ab} and X_{cd} will be said to be equal if whenever the circuit a-b is open, the circuit c-d is open, and whenever a-b is closed, c-d is closed. Now let the symbol + (plus) be defined to mean the series connection of the two terminal circuits whose hinderances are added together. Thus $X_{ab} + X_{cd}$ is the hinderance of the circuit a-d when b and c are connected together. Similarly the product of two hinderances $(X_{ab} \cdot X_{cd})$ will be defined to mean the

hinderance of the circuit formed by connecting the circuits a-b and c-d in parallel. A relay contact or switch will be represented in a circuit by the symbol in Fig. 1, the letter being the corresponding hinderance function. Fig. 2 shows the interpretation of the plus sign and Fig. 3 the multiplication sign.

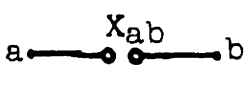


Fig. 1

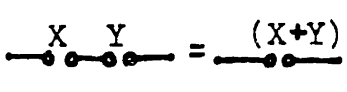


Fig. 2

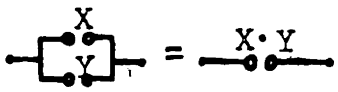


Fig. 3

This choice of symbols makes the manipulation of hinderances very similar to ordinary numerical algebra.

It is evident that with the above definitions the following postulates will hold:

Postulates

1. a.	$0 \cdot 0 = 0$	A closed circuit in parallel with a closed circuit is a closed circuit.
b.	$1 + 1 = 1$	An open circuit in series with an open circuit is an open circuit.
2. a.	$1 + 0 = 0 + 1 = 1$	An open circuit in series with a closed circuit in either order is an open circuit.
b.	$0 \cdot 1 = 1 \cdot 0 = 0$	A closed circuit in parallel with an open circuit in either order is a closed circuit.

3.	a. $0 + 0 = 0$	A closed circuit in series with a closed circuit is a closed circuit.
	b. $1 \cdot 1 = 1$	An open circuit in parallel with an open circuit is an open circuit.
4.		At any given time either $X = 0$ or $X = 1$.

These are sufficient to develop all the theorems which will be used in connection with circuits containing only series and parallel connections. The postulates are arranged in pairs to emphasize a duality relationship between the operations of addition and multiplication and the quantities zero and one. Thus if in any of the 'a' postulates the zero's are replaced by one's and the multiplications by additions and vice versa, the corresponding 'b' postulate will result. This fact is of great importance. It gives each theorem a dual, it being necessary to prove only one to establish both. The only one of these postulates which differs from ordinary algebra is 1b. However, this enables great simplifications in the manipulation of these symbols.

Theorems. In this section a number of theorems governing the combination of hinderances will be given. Inasmuch as any of the theorems may be proved by a very simple process, the proofs will not be given

except for an illustrative example. The method of proof is that of "perfect induction," i.e., the verification of the theorem for all possible cases. Since by postulate 4 each variable is limited to the values 0 and 1, this is a simple matter. Some of the theorems may be proved more elegantly by recourse to previous theorems, but the method of perfect induction is so universal that it is probably to be preferred.

1. a. $x + y = y + x$

   b. $xy = yx$

2. a. $x + (y + z) = (x + y) + z$

   b. $x(yz) = (xy)z$

3. a. $x(y + z) = xy + xz$

   b. $x + yz = (x + y)(x + z)$

4. a. $1 \cdot x = x$

   b. $0 + x = x$

5. a. $1 + x = 1$

   b. $0 \cdot x = 0$

For example, to prove theorem 4a, note that X is either 0 or 1. If it is 0, the theorem follows from postulate 2b; if 1, it follows from postulate 3b.

We shall now define a new operation to be called negation. The negative of a hinderance X will be written X' and is defined as a variable which is equal to 1 when X equals 0 and equal to 0 when X

equals 1. If X is the hinderance of the make contacts of a relay, then X' is the hinderance of the break contacts of the same relay. The definition of the negative of a hinderance gives the following theorems:

6. a. $X + X' = 1$

   b. $XX' = 0$

7. a. $0' = 1$

   b. $1' = 0$

8. $(X')' = X$

Analogue with the Calculus of Propositions. We are now in a position to demonstrate the equivalence of this calculus with certain elementary parts of the calculus of propositions. The algebra of logic (1), (2), (3) originated by George Boole, is a symbolic method of investigating logical relationships. The symbols of Boolean algebra admit of two logical interpretations. If interpreted in terms of classes, the variables are not limited to the two possible values 0 and 1. This interpretation is known as the algebra of classes. If, however, the terms are taken to represent propositions, we have the calculus of propositions in which variables are limited to the values 0 and 1*,

*This refers only to the classical theory of the Calculus of Propositions. Recently some work has been done with logical systems in which propositions may have more than two "truth values."

as are the hinderance functions above. Usually the two subjects are developed simultaneously from the same set of postulates, except for the addition in the case of the Calculus of Propositions of a postulate equivalent to postulate 4 above. E.V. Huntington (4) gives the following set of postulates for symbolic logic:

1. The class K contains at least two distinct elements.
2. If a and b are in the class K then a + b is in the class K.
3. $a + b = b + a$
4. $(a + b) + c = a + (b + c)$
5. $a + a = a$
6. $ab + ab' = a$ where ab is defined as $(a' + b')'$

If we let the class K be the class consisting of the two elements 0 and 1, then these postulates follow from those given on pages 5 and 6. Also postulates 1, 2, and 3 given there can be deduced from Huntington's postulates. Adding 4 and restricting our discussion to the calculus of propositions, it is evident that a perfect analogy exists between the calculus for switching circuits and this branch of symbolic logic.* The two interpretations of the symbols are shown in Table 1.

---

*This analogy may also be seen from a slightly different viewpoint. Instead of associating X_{ab} directly with the circuit a-b let X_{ab} represent the <u>proposition</u> that the

Due to this analogy any theorem of the Calculus of Propositions is also a true theorem if interpreted in terms of relay circuits. The remaining theorems in this section are taken directly from this field.

De Morgan's theorem:

9. a. $(X + Y + \ldots)' = X'.Y'.Z'\ldots$

b. $(X.Y.Z \ldots)' = X' + Y' + Z' + \ldots$

This theorem gives the negative of a sum or product in terms of the negatives of the summands or factors. It may be easily verified for two terms by substituting all possible values and then extended to any number n of variables by mathematical induction.

A function of certain variables $X_1, X_2 \ldots . X_n$ is any expression formed from the variables with the operations of addition, multiplication, and negation. The notation $f(X_1, X_2, \ldots X_n)$ will be used to represent a function. Thus we might have $f(X, Y, Z)) = XY + X' (Y' + Z')$. In infinitesimal calculus it is shown that any function (providing it is continuous and all derivatives are continuous) may be expanded in a Taylor Series. A somewhat similar expansion is possible in the Calculus of propositions. To develop the series expansion of functions

---

(Footnote continued from preceding page)
circuit a-b is open. Then all the symbols are directly interpreted as propositions and the operations of addition and multiplication will be seen to represent series and parallel connections.

TABLE I

Analogue Between the Calculus of Propositions and the Symbolic Relay Analysis

Symbol	Interpretation in relay circuits	Interpretation in the Calculus of Propositions
X	The circuit X.	The proposition X.
0	The circuit is closed.	The proposition is false.
1	The circuit is open.	The proposition is true.
X + Y	The series connection of circuits X and Y	The proposition which is true if either X or Y is true.
XY	The parallel connection of circuits X and Y	The proposition which is true if both X and Y are true.
X'	The circuit which is open when X is closed, and closed when X is open.	The contradictory of proposition X.
=	The circuits open and close simultaneously.	Each proposition implies the other.

first note the following equations:

10. a. $f(X_1, X_2, \ldots X_n) = X_1 f(1, X_2 \ldots X_n) + X_1' f(0, X_2 \ldots X_n)$

b. $f(X_1 \ldots X_n) = [f(0, X_2 \ldots X_n) + X_1] \cdot [f(1, X_2 \ldots X_n) + X_1']$

These reduce to identities if we let X_1 equal either 0 or 1. In these equations the function f is said to be expanded about X_1. The coefficients of X_1 and X_1' in 10a are functions of the (n-1) variables $X_2 \ldots X_n$ and may thus be expanded about any of these variables in the same manner. The additive terms in 10b also may be expanded in this manner. Thus we get:

11. a. $f(X_1 \ldots X_n) = X_1 X_2 f(1,1,X_3 \ldots X_n) + X_1 X_2' f(1,0,X_3 \ldots X_n) + X_1' X_2 f(0,1,X_3 \ldots X_n) + X_1' X_2' f(0,0,X_3 \ldots X_n)$

b. $f(X_1 \ldots X_n) = [X_1 + X_2 + f(0,0,X_3 \ldots X_n)] \cdot [X_1 + X_2' + f(0,1,\ldots X_n)] \cdot [X_1' + X_2 + f(1,0, \ldots X_n)] \cdot [X_1' + X_2' + f(1,1,X_3 \ldots X_n)]$

Continuing this process n times we will arrive at the complete series expansion having the form:

12. a. $f(X \ldots X_n) = f(1,1,1 \ldots 1) X_1 X_2 \ldots X_n + f(0,1, 1 \ldots 1) X_1' X_2 \ldots X_n + \ldots\ldots + f(0,0,0 \ldots 0) X_1' X_2' \ldots X_n'$

b. $f(X_1 \ldots X_n) = [X_1 + X_2 + \ldots X_n + f(0,0,0 \ldots 0)] \cdot [X_1' + X_2 \ldots + X_n + f(1,0,0 \ldots 0)] \cdot \ldots\ldots [X_1' + X_2' + \ldots X_n' + f(1,1, \ldots 1)]$

By 12a, f is equal to the sum of the products formed by permuting primes on the terms of $X_1X_2....X_n$ in all possible ways and giving each product a coefficient equal to the value of the function when that product is 1. Similarly for 12b.

As an application of the series expansion it should be noted that if we wish to find a circuit representing any given function we can always expand the function by either 10a or 10b in such a way that any given variable appears at most twice, once as a make contact and once as a break contact. This is shown in Fig. 4.

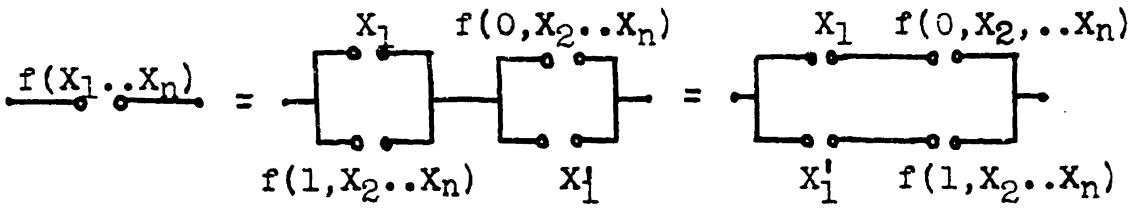


Fig. 4

Similarly by 11 any other variable need appear no more than 4 times (two make and two break contacts) etc.

A generalization of De Morgan's theorem is represented symbolically in the following equation:

$$13.\ [f(X_1,X_2,...X_n,+,\cdot)]' = f(X_1',X_2',....X_n',\cdot,+)$$

By this we mean that the negative of any function may

be obtained by replacing each variable by its negative and interchanging the + and • symbols. Explicit and implicit parentheses will, of course, remain in the same places. For example, the negative of X + Y•(Z + WX') will be X'(Y' + Z'(W' + X)).

Some other theorems useful in simplifying expressions are given below:

14. a. $X = X + X = X + X + X = \text{etc.}$

    b. $X = X \cdot X = X \cdot X \cdot X = \text{etc.}$

15. a. $X + XY = X$

    b. $X(X + Y) = X$

16. a. $XY + X'Z = XY + X'Z + YZ$

    b. $(X + Y)(X' + Z) = (X + Y)(X' + Z)(Y + Z)$

17. a. $Xf(X) = Xf(1)$

    b. $X + f(X) = X + f(0)$

18. a. $X'f(X) = X'f(0)$

    b. $X' + f(X) = X' + f(1)$

Any expression formed with the operations of addition, multiplication, and negation represents explicitly a circuit containing only series and parallel connections. Such a circuit will be called a series-parallel circuit. Each letter in an expression of this sort represents a make or break relay contact, or a switch blade and contact. To find the circuit requiring the least number of contacts, it is

therefore necessary to manipulate the expression into the form in which the least number of letters appear. The theorems given above are always sufficient to do this. A little practice in the manipulation of these symbols is all that is required. Fortunately most of the theorems are exactly the same as those of numerical algebra--the associative, commutative, and distributive laws of algebra hold here. The writer has found theorems 3, 6, 9, 14, 15, 16a, 17, and 18 to be especially useful in the simplification of complex expressions.

As an example of the simplification of expressions consider the circuit shown in Fig. 5.

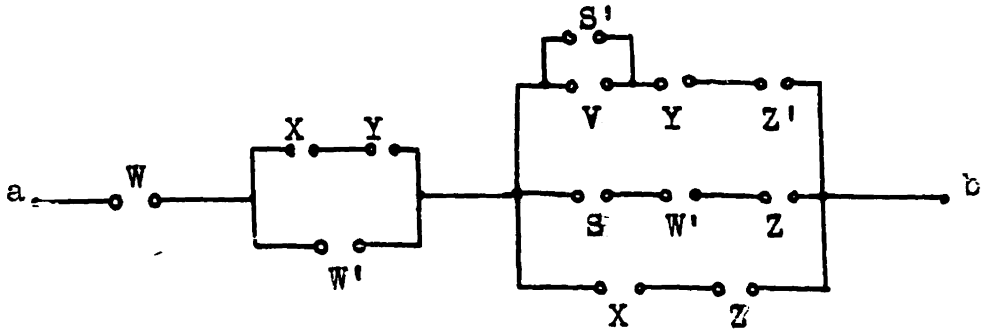


Fig. 5

The hinderance function X_{ab} for this circuit will be:

$$X_{ab} = W+W'(X+Y) + (X+Z)(S+W'+Z)(Z'+Y+S'V)$$
$$= W+X+Y+(X+Z)(S+W'+Z)(Z'+Y+S'V)$$
$$= W+X+Y+Z(Z'+S'V)$$

These reductions were made with 17b using first W, then X and

Y as the "X" of 17b. Now multiplying out:

$$X_{ab} = W + X + Y + ZZ' + ZS'V$$
$$= W + X + Y + ZS'V$$

The circuit corresponding to this expression is shown in Fig. 6. Note the large reduction in the number of elements.

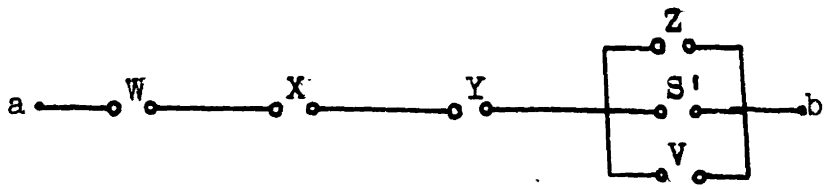


Fig. 6

It is convenient in drawing circuits to label a relay with the same letter as the hinderance of make contacts of the relay. Thus if a relay is connected to a source of voltage through a network whose hinderance function is X, the relay and any make contacts on it would be labeled X. Break contacts would be labeled X'. This assumes that the relay operates instantly and that the make contacts close and the break contacts open simultaneously. Cases in which there is a time delay will be treated later.

It is also possible to use the analogy between Boolean algebra and relay circuits in the opposite direction, i.e., to represent logical relations by

means of electric circuits. Some interesting results have been obtained along this line, but are of no importance here.

III Multi-terminal and Non-series-parallel Circuits

Equivalence of n-Terminal Networks. The usual relay control circuit will take the form of Fig. 7, where $X_1, X_2, \ldots X_n$ are relays or other devices controlled by the circuit and N is a network of relay contacts and switches.

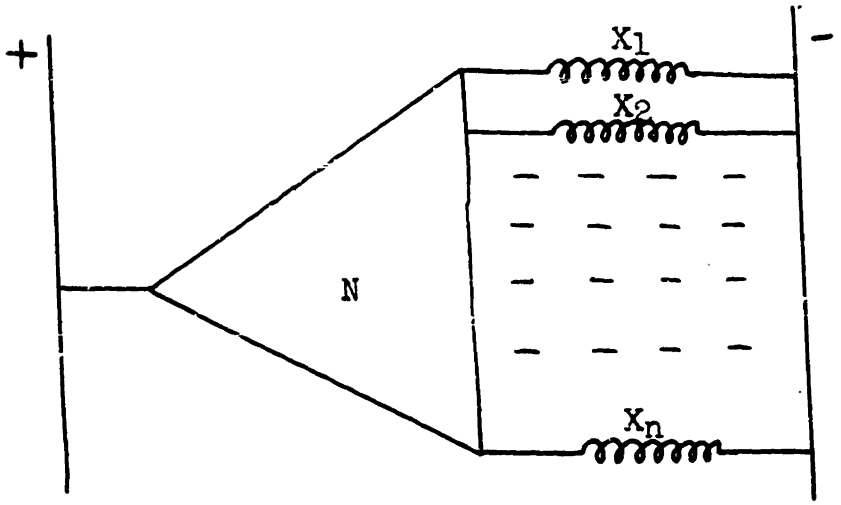


Fig. 7

It is desirable to find transformations that may be applied to N which will keep the operation of all the relays $X_1 \ldots X_n$ the same. So far we have only considered transformations which may be applied to a two-terminal network keeping the operation of one relay in series with this network the same. To this end we shall define equivalence of two n-terminal networks as follows:

Definition: Two n-terminal networks M and N will be said to be equivalent with respect to these

terminals if and only if $X_{jk} = Y_{jk}$ j, k = 1, 2, 3,...n where X_{jk} is the hinderance on network N between terminals j and k, and Y_{jk} is that for M between the corresponding terminals.

Thus under this definition the equivalences of the preceding sections were with respect to two terminals.

Star-Mesh and Delta-Wye Transformations. As in ordinary network theory there exist star to mesh and delta to wye transformations. The delta to wye transformation is shown in Fig. 8. These two networks are equivalent with respect to the three terminals a, b, and c, since by the distributive law $X_{ab} = R(S + T) = RS + RT$ and similarly for the other pairs of terminals a-c and b-c.

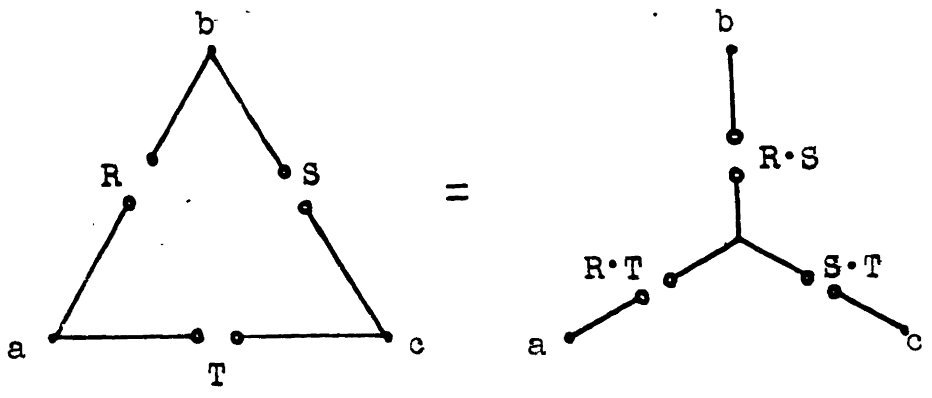


Fig. 8

The wye to delta transformation is shown in Fig. 9. This follows from the fact that $X_{ab} = R + S = (R + S)(R + T + T + S)$.

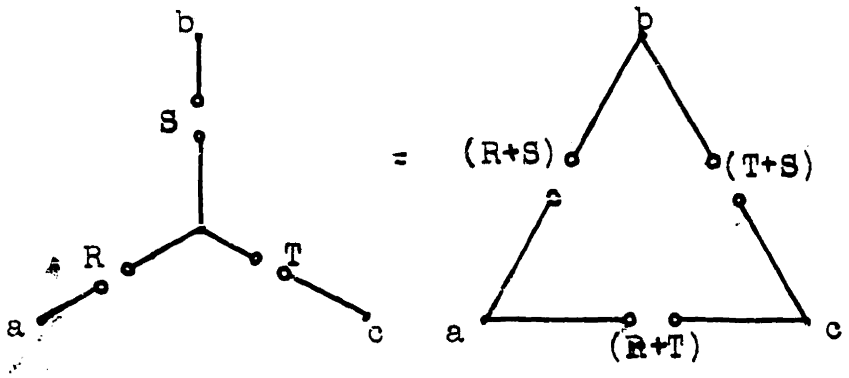


Fig. 9

An n point star also has a mesh equivalent with the central node eliminated. This is formed exactly as in the simple three point star, by connecting each pair of terminals of the mesh through a hinderance which is the sum of the corresponding arms of the star. For n = 5 this is shown in Fig. 10.

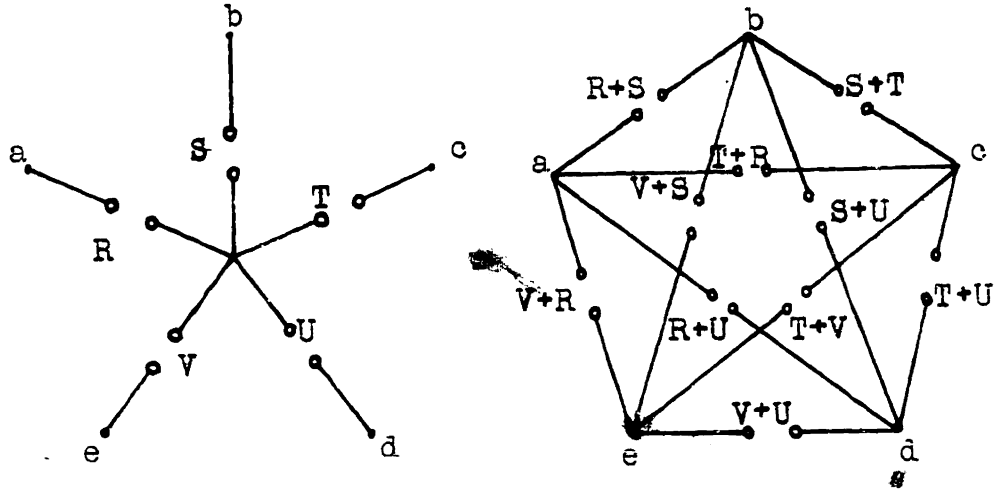


Fig. 10

Hinderance Function of a Non-Series-Parallel Network.

The methods of Part II were not sufficient to handle circuits which contained connections other than those of a series-parallel type. The bridge of Fig. 11, for example, is a non-series-parallel network. These networks will be handled by reducing to an equivalent series-parallel circuit. Three methods have been developed for finding the equivalent of a network such as the bridge.

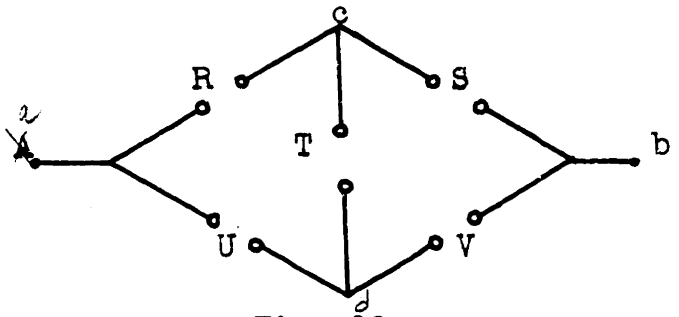


Fig. 11

The first is the obvious method of applying the transformations until the network is of the series-parallel type and then writing the hinderance function by inspection. This process is exactly the same as is used in simplifying complex impedance networks. To apply this to the circuit of Fig. 11, first eliminate the node c, by applying the star to mesh transformation to the star a-c, b-c, d-c. This gives the network of Fig. 12.

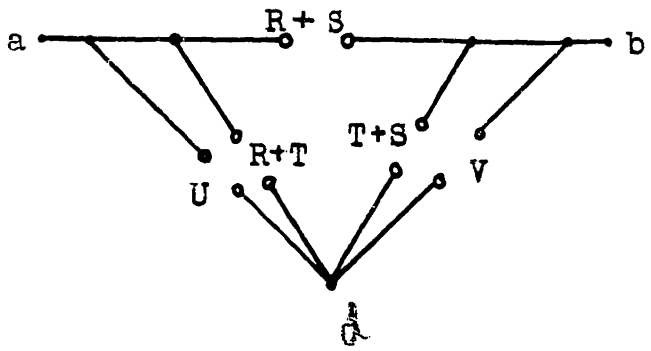


Fig. 12

The hinderance function may be written down from inspection for this network.

$$X_{ab} = (R + S)[U(R + T) + V(T + S)]$$

Simplifying by the theorems gives:

$$X_{ab} = RU + SV + RTV + STU$$

The second method of analysis is to draw all possible paths between the points under consideration through the network. These paths are drawn along the lines representing the component hinderance elements of the circuit. If any one of these paths has zero hinderance, the required function must be zero. Hence if the result is written as a product, the hinderance of each path will be a factor of this product. The required result may therefore be written as the product of the hinderances of all possible paths between the two points. Paths which touch the same point more than once need

not be considered. In Fig. 13 this method is applied to the bridge. The paths are marked in red.

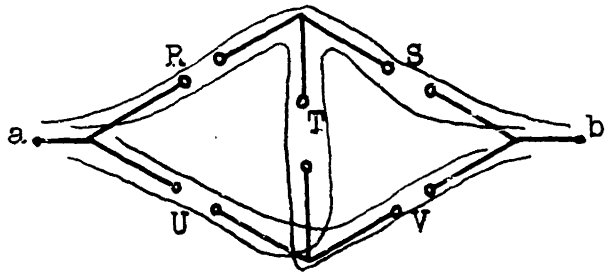


Fig. 13

The function is therefore given by:

$$X_{ab} = (R + S)(U + V)(R + T + V)(U + T + S)$$
$$= RU + SV + RTV + UTS$$

The same result is thus obtained as with the first method.

The third method, the dual of the second, is to draw all possible lines which would break the circuit between the points under consideration, making the lines go through the hinderances of the circuit. The result is written as a sum, each term corresponding to a certain line. These terms are the products of all the hinderances on the line. This method is applied to the bridge in Fig. 14, the lines being drawn in red.

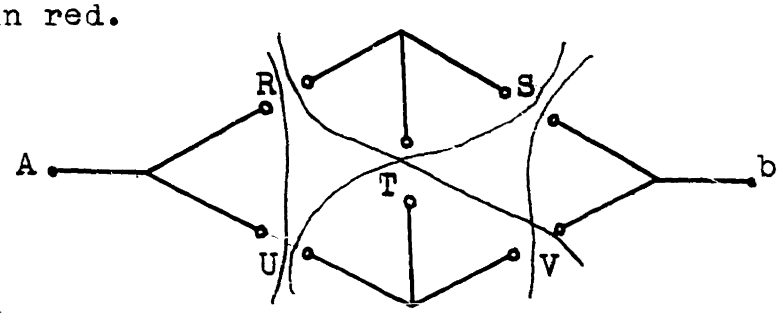


Fig. 14

This again gives for the hinderance of the network:

$$X_{ab} = RUt + SV + RTV + STU$$

The third method is usually the most convenient and rapid, for it gives the result directly as a sum. It seems much easier to handle sums than products due, no doubt, to the fact that in ordinary algebra we have the distributive law $X(Y + Z) = XY + XZ$, but not its dual $X + YZ = (X + Y)(X + Z)$. It is, however, sometimes difficult to apply the third method to non-planar networks (networks which cannot be drawn on a plane without crossing lines) and in this case one of the other two methods may be used.

Simultaneous Equations. If there are n dependent variables, there will be n simultaneous equations defining the system. Any additive terms which are common to several of the functions may be factored out in the manner illustrated by the following example. These terms need only be realized once to take care of all the functions in which they appear.

$$W = A + B + CW$$
$$X = A + B + WX$$
$$Y = A + CY$$
$$Z = EZ + f$$

$$\left.\begin{matrix} W = \\ X = \\ Y = \end{matrix}\right| A + \left| \begin{matrix} B + \left|\begin{matrix} CW \\ WX \end{matrix}\right. \\ CY \end{matrix}$$
$$Z = EZ + f$$

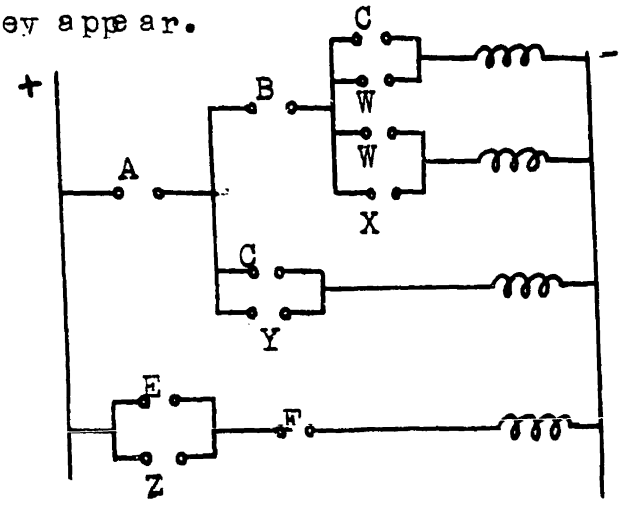


Fig. 15

Sometimes the relation $ab' = 0$ obtains between two relays a and b. This is true, for example, in a sequential system where each relay of the sequence locks itself in and a precedes b in the sequence. Whenever b is operated a is operated. In such a case the following simplifications may be made:

If $ab' = 0$

Then $a'b' = a'b' + ab' = b'$

$ab = ab + ab' = a$

$a' + b = 1$

$(a' + b') = (a' + b')(a' + b) = a'$

$(a + b) = (a + b)(a' + b) = b$

Matrix Methods. It is also possible to treat multi-terminal networks by means of matrices. Although useful for theoretical work the method is cumbersome for practical problems and will therefore only be briefly sketched. We shall assume the same rules of manipulation of matrices as usually defined in works on higher algebra, the only difference being that the elements of our matrices will be hinderance functions rather than ordinary algebraic numbers or variables. The X' matrix of a network with n nodes will be defined as the following array:

$$\left\| \begin{array}{cccc} 1 & X'_{12} & X'_{13} & \ldots\ldots\ldots\ldots X'_{1n} \\ X'_{21} & 1 & X'_{23} & \ldots\ldots\ldots\ldots X'_{2n} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ X'_{n1} & X'_{n2} & \ldots\ldots\ldots\ldots & 1 \end{array} \right\|$$

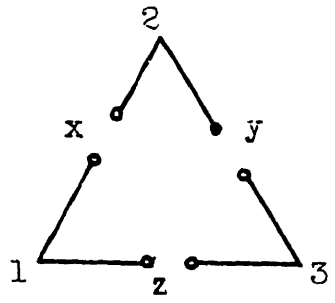
where X'_{jk} is the negative of the hinderance common to nodes j and k.

Theorem: The X' matrix of a network formed by connecting two n node networks in parallel (corresponding nodes together) is the sum of the individual X' matrices. This theorem is more general than might appear at first since any interconnection of two networks may be considered as a parallel connection of two networks with the same number of nodes by adding nodes as needed whose mutual hinderances to the other nodes is one.

Now define a matrix to be called the U' matrix of a network as follows:

$$\left\| \begin{array}{cccc} 1 & U_{12} & \ldots\ldots & U_{1n} \\ U_{21} & 1 & \ldots\ldots & U_{2n} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ U_{n1} & U_{n2} & \ldots\ldots & 1 \end{array} \right\|$$

where U'_{jk} is the negative of the hinderance function from node j to k, the network considered as a two terminal circuit. Thus for the three node network of Fig. 16 the X' and U' matrices are as shown at the right.



$$\begin{Vmatrix} 1 & x' & z' \\ x' & 1 & y' \\ z' & y' & 1 \end{Vmatrix} \qquad \begin{Vmatrix} 1 & x'+y'z' & z'+x'y' \\ x'+y'z' & 1 & y'+x'z' \\ z'+x'y' & y'+x'z' & 1 \end{Vmatrix}$$

Fig. 16 X' Matrix U' Matrix

Theorem: Any power of the X' matrix of a network gives a network which is equivalent with respect to all nodes. The matrix is raised to a power by the usual rule for multiplication of matrices.

Theorem:

$$\begin{Vmatrix} 1 & U'_{12} & \cdots & U'_{1n} \\ U'_{21} & 1 & \cdots & U'_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ U'_{n1} & \cdots & \cdots & 1 \end{Vmatrix} = \begin{Vmatrix} 1 & X'_{12} & \cdots & X'_{1n} \\ X'_{21} & 1 & \cdots & X'_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ X'_{1n} & \cdots & \cdots & 1 \end{Vmatrix}^{s} \qquad s \geq n-1$$

Theorem: Any node, say the kth, may be eliminated leaving the network equivalent with respect to all remaining nodes by adding to each element X'_{rs} of the

X' matrix the product of the elements X'_{rk} and X'_{ks}, and striking out the kth row and column.

Thus eliminating the 3rd node of Fig. 16 we get:

$$\begin{Vmatrix} 1+z'z' & x'+z'y' \\ x'+y'z' & 1+y'y' \end{Vmatrix} = \begin{Vmatrix} 1 & x'+y'z' \\ x'+y'z' & 1 \end{Vmatrix}$$

The proofs of these theorems are of a simple nature, but quite long and will not be given.

Special Types of Relays and Switches. In certain types of circuits it is necessary to preserve a definite sequential relation in the operation of the contacts of a relay. This is done with make-before-break (or continuity) and break-make (or transfer) contacts. In handling this type of circuit the simplest method seems to be to assume in setting up the equations that the make and break contacts operate simultaneously, and after all simplifications of the equations have been made and the resulting circuit drawn the required type of contact sequence is found from inspection.

Relays having a time delay in operating or deoperating may be treated similarly or by shifting

the time axis. Thus if a relay coil is connected to a battery through a hinderance X, and the relay has a delay of p seconds in operating and releasing, then the hinderance function of the contacts of the relay will also be X, but at a time p seconds later. This may be indicated by writing X(t) for the hinderance in series with the relay, and X(t-p) for that of the relay contacts.

There are many special types of relays and switches for particular purposes, such as the stepping switches and selector switches of various sorts, multi-winding relays, cross-bar switches, etc. The operation of all these types may be described with the words "or," "and," "if," "operated," and "not operated." This is a sufficient condition that may be described in terms of hinderance functions with the operations of addition, multiplication, negation, and equality. Thus a two winding relay might be so constructed that it is operated if the first or the second winding is operated (activated) and the first and the second windings are not operated. Usually, however, these special relays occur only at the end of a complex circuit and may be omitted entirely from the calculations to be added after the rest of the circuit is designed.

Sometimes a relay X is to operate when a circuit r closes and to remain closed independent of r until a circuit S opens. Such a circuit is known as a lock-in circuit. Its equation is:

$$X = rX + S$$

Replacing X by X' gives:

$$X' = rX' + S$$

or

$$X = (r' + X)S'$$

In this case X is opened when r closes and remains open until S opens.

IV Synthesis of Networks

Some General Theorems on Networks and Functions. It has been shown that any function may be expanded in a series consisting of a sum of products, each product being of the form $X_1X_2\ldots X_n$ with some permutation of primes on the letters, and each product having the coefficient 0 or 1. Now since each of the n variables may or may not have a prime, there is a total of 2^n different products of this form. Similarly each product may have the coefficient 0 or the coefficient 1 so there are 2^{2^n} possible sums of this sort. Each of these sums will represent a unique function, but some of the functions may actually involve less than n variables (i.e., they are of such a form that for one or more of the n variables, say X_k, we have identically $f(X_1,\ldots X_{k-1}, 0, X_{k+1},\ldots X_n) = f(X_1\ldots X_{k-1}, 1, X_{k+1}, \ldots X_n)$ so that under no conditions does the value of the function depend on the value of X_k.

Hence we have the theorem:

Theorem: The number of functions of n variables or less is 2^{2^n}.

To find the number of functions which actually involve n variables we proceed as follows. Let $\phi(n)$ be the required number. Then by the theorem just given:

$$2^{2^n} = \sum_{k=0}^{n} \binom{n}{k} \phi(k)$$

where $\binom{n}{k} = n!/k!(n-k)!$ is the number of combinations of n things taken k at a time. Solving for $\phi(n)$ gives:

$$\phi(n) = 2^{2^n} - \sum_{k=0}^{n-1} \binom{n}{k} \phi(k)$$

By substituting for $\phi(n-1)$ on the right the similar expression found by replacing n by n-1 in this equation, x then similarly substituting for $\phi(n-2)$ in the expression thus obtained, etc, an equation may be obtained involving only $\phi(n)$. This equation may then be simplified to the form:

$$\phi(n) = \sum_{k=0}^{n} \left[\binom{n}{k} 2^{2^k} (-1)^{n-k}\right]$$

As n increases this expression approaches its leading term 2^{2^n} asymptotically. The error in using only this term for n = 5 is less than .01%.

We shall now determine those functions of n variables which require the most relay contacts to realize, and find the number of contacts required. In order to do this, it is necessary to define a function of two variables known as the sum modulo two or disjunct of the variables. This function is written $X_1 \oplus X_2$ and is defined by the equation:

$$X_1 \oplus X_2 = X_1 X_2' + X_1' X_2$$

It is easy to show that the sum modulo two obeys the commutative, associative, and the distributive law with respect to multiplication, i.e.

$$X_1 \oplus X_2 = X_2 \oplus X_1$$

$$(X_1 \oplus X_2) \oplus X_3 = X_1 \oplus (X_2 \oplus X_3)$$

$$X_1(X_2 \oplus X_3) = X_1X_2 \oplus X_1X_3$$

Also:

$$(X_1 \oplus X_2)' = X_1 \oplus X_2'$$

$$X_1 \oplus 0 = X_1$$

$$X_1 \oplus 1 = X_1'$$

Since the sum modulo two obeys the associative law, we may omit parentheses in a sum of several terms without ambiguity. The sum modulo two of the n variables $X_1, X_2 \ldots X_n$ will for convenience be written:

$$X_1 \oplus X_2 \oplus X_3 \ldots \oplus X_n = \sum_{k=1}^{n}{}_2 X_k$$

Theorem: The two functions of n variables which require the most elements (relay contacts) in a series-parallel realization are $\sum_{1}^{n}{}_2 X_k$ and $(\sum_{1}^{n}{}_2 X_k)'$, each of which requires $(3 \cdot 2^{n-1} - 2)$ elements.

This will be proved by mathematical induction. First note that it is true for n = 2. There are 10 functions of 2 variables, namely, XY, X+Y, X'Y, X'+Y, XY', X+Y', X'Y', X'+Y', XY' + X'Y, XY+X'Y'. All of these but the last two require two elements; the last two require four elements and are X⊕Y and (X⊕Y)' respectively. Thus the theorem is true for n = 2. Now assuming it true for n-1, we shall prove it true for n and thus complete the induction. Any function of n variables may be written by (10a):

$$f(X_1,X_2,\ldots X_n) = X_1 f(1,X_2,\ldots X_n) + X_1' f(0,X_2\ldots X_n) \qquad (19)$$

Now the terms $f(1,X_2\ldots X_n)$ and $f(0,X_2\ldots X_n)$ are functions of n-1 variables, and if they individually require the most elements for n-1 variables, then f will require the most elements for n variables, providing there is no other method of writing f so that less elements are required. We have assumed that the most elements for these n-1 variables are required by $\sum_2^n X_k$ and $(\sum_2^n X_k)'$. If we therefore substitute for $f(1,X_2\ldots X_n)$ the function $\sum_2^n X_k$ and for $f(0,X_2..X_n)$ the function $(\sum_2^n X_k)'$ we get:

$$f = X_1 \sum_2^n X_k + X_1'(\sum_2^n X_k)' = (\sum_1^n X_k)'$$

From the symetry of this function there is no other way of expanding which will reduce the number of elements. If the functions are substituted in the other order, we get:

$$f = X_1(\sum_2^n X_k)' + X_1'\sum_2^n X_k = \sum_1^n X_k$$

This completes the proof that these functions require the most elements. To show that each requires $(3\cdot 2^{n-1}-2)$ elements, let the number of elements required be denoted by s(n). Then from (19) we get the difference equation:

$$s(n) = 2s(n-1) + 2$$

with $s(2) = 4$. This is linear, with constant coefficients, and may be solved by the usual methods (5). The solution is:

$$s(n) = 3\cdot 2^{n-1}-2$$

as may be easily verified by substituting in the difference equation and boundary condition.

Note that the above only applies to a series-parallel realization. In a later section it will be shown that the function $\sum_1^n X_k$ and its negative may be realized with 4(n-1) elements using a more general type of circuit. The function requiring the most elements using any type of circuit has not as yet been determined.

Dual Networks. The negative of any network may be found by De Morgan's theorem, but the network must first be transformed into an equivalent series-parallel circuit (unless it is already of this type). A theorem will be developed with which the negative of any planar two-terminal circuit may be found directly. As a corollary a method of finding a constant current circuit equivalent to a given constant voltage circuit and vice versa will be given.

Let N represent a planar network of hinderances, with the function X_{ab} between the terminals a and b which are on the outer edge of the network. For definiteness consider the network of Fig. 17 (here the hinderances are shown merely as lines). Now let M represent the dual of N, as found by the following process; for each contour or mesh of N assign a node or junction point of M. For each element of N say X_k, separating the contours r and s there corresponds an element X_k' connecting the nodes r and s of M. The area exterior to N is to be considered as two meshes, c and d, corresponding to nodes c and d of M. Thus the dual of Fig. 17 is the network of Fig. 18.

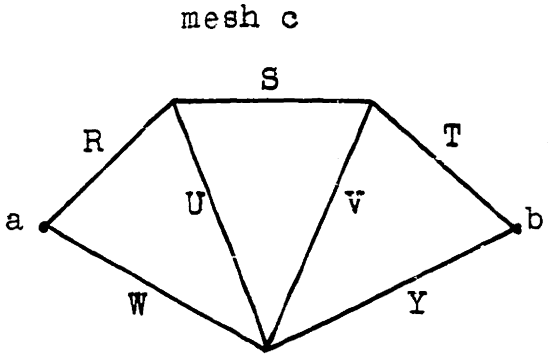


Fig. 17

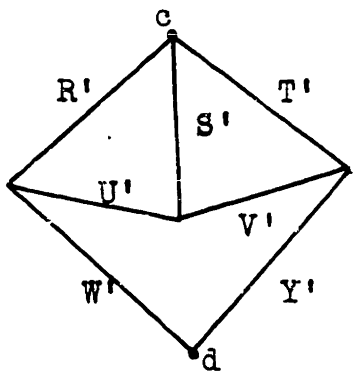


Fig. 18

Theorem: If M and N bear this duality relationship, then $X_{ab} = X'_{cd}$.

To prove this, let the networks M and N be superimposed, the nodes of M within the corresponding meshes of M and corresponding elements crossing. For the network of Fig. 17, this is shown in Fig. 19, with N in black and M in red. Incidentally, the easiest method of finding the dual of a network (whether of this type or an impedance network) is to draw the required network superimposed on the given network. Now, if $X_{ab} = 0$, then there must be some path from a to b along the lines of N such that every element on this path equals zero. But this path represents a path across M dividing the circuit from c to d along which every element of M is one. Hence $X_{cd} = 1$. Similarly, if $X_{cd} = 0$, then $X_{ab} = 1$, and it follows that $X_{ab} = X'_{cd}$.

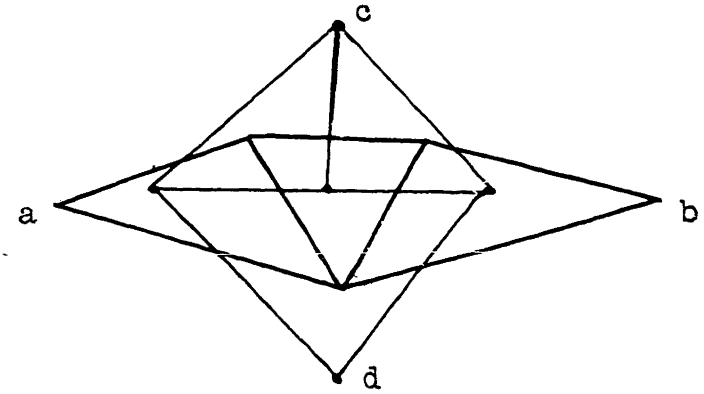


Fig. 19

In a constant-voltage relay system all the relays are in parallel across the line. To open a relay a series connection is opened. The general constant-voltage system is shown in Fig. 20. In a constant-current system the relays are all in series in the line. To de-operate a relay, it is short circuited. The general constant-current circuit corresponding to Fig. 20 is shown in Fig. 21. If the relay Y_k of Fig. 21 is to be operated whenever the relay X_k of Fig. 20 is operated and not otherwise, then evidently the hinderance in parallel with Y_k which shorts it out must be the negative of the hinderance in series with X_k which connects it across the voltage source. If this is true for all the relays, we shall say that the constant-current and constant-voltage systems are equivalent. The above theorem may be used to find equivalent circuits of this sort. For, if we make the networks N and M of Figs. 20 and 21 duals in the sense described, then the condition will be satisfied.

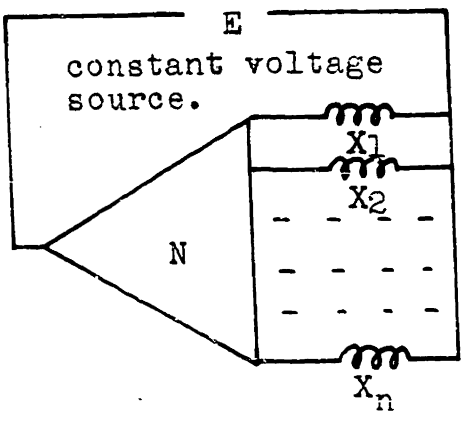


Fig. 20

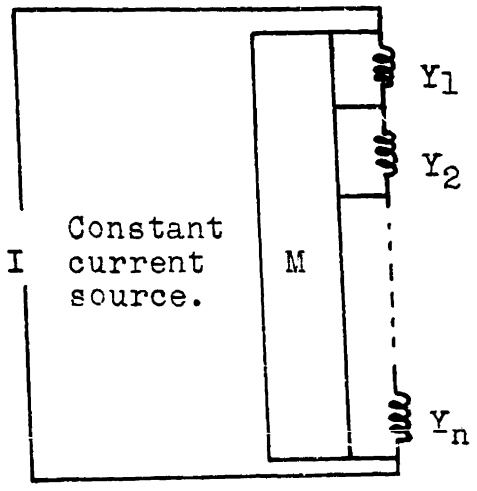


Fig. 21

A simple example of this is shown in Figs. 22 and 23.

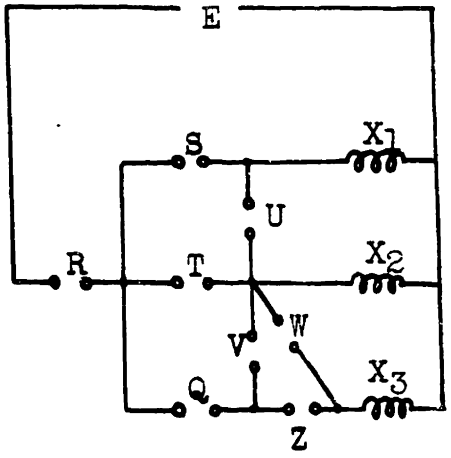


Fig. 22

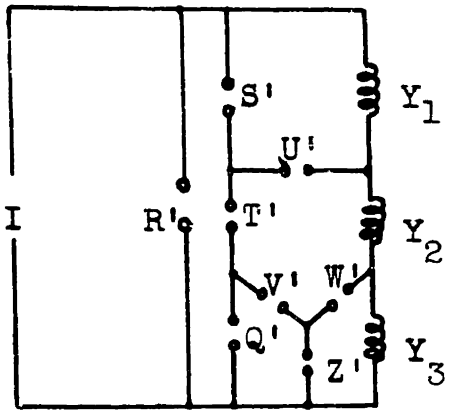


Fig. 23

Synthesis of the General Symetric Function. As has been shown, any function represents explicitly a series-parallel circuit. The series-parallel realization may require more elements, however, than some other circuit representing the same function. In this section a method will be given for finding a circuit representing a certain type of function which in general is much more economical of elements than the best series-parallel circuit. This type of function frequently appears in relay circuits and is of much importance.

A function of the n variables $X_1, X_2, \ldots X_n$ is said to be symmetric in these variables if any interchange of these variables leaves the function

identically the same. Thus XY + XZ + YZ is symmetric in the variables X, Y, and Z. Since any permutation of variables may be obtained by successive interchanges of two variables, a necessary and sufficient condition that a function be symmetric is that any interchange of two variables leaves the function unaltered.

We now give a theorem which forms the basis of the method of synthesis to be described.

Theorem: The necessary and sufficient condition that a function be symmetric is that it may be specified by stating a set of numbers $a_1, a_2, \ldots . a_k$ such that if exactly a_j ($j = 1, 2, 3, \ldots k$) of the variables are zero, then the function is zero and not otherwise. This follows easily from the definition. For the example given these numbers are 2 and 3.

Theorem: There are 2^{n+1} symmetric functions of n variables. For every selection of a set of numbers from the numbers 0, 1, 2, ...n corresponds to one and only one symmetric function. Since there are n+1 numbers each of which may be either taken or not in our selection, the total number of functions is 2^{n+1}. Two of these functions are trivial, however, namely the selections in which none and all of the numbers are taken. These give the "functions"1 and 0 respectively.

By proper selection of the varialbes many apparently unsymmetric functions may be made symmetric. For example, XY'Z + X'YZ + X'Y'Z', although not symmetric in X, Y, and Z, is symmetric in X, Y, and Z'.

The set of numbers $a_1, a_2, \ldots a_k$ will for convenience be called the a-numbers of the function. The theorems concerning combinations of symmetric functions are most easily stated in terms of the classes of a-numbers. For this reason we denote the class of a-numbers by a single letter A. If two different sets of a-numbers are under consideration they will be denoted by A_1 and A_2. The symmetric function of n variables having the a-numbers $a_1, a_2 \ldots a_k$ will be written $S_n(a_1, a_2 \ldots a_k)$ or $S_n(A)$.

Theorem: $S_n(A_1) \cdot S_n(A_2) = S_n(A_1 + A_2)$

where $A_1 + A_2$ means the <u>logical sum</u> of the classes A_1 and A_2 i.e., the class of those numbers which are members of either A_1 or A_2 or both. Thus $S_6(1, 2, 3) \cdot S_6(2, 3, 5)$ is equal to $S_6(1, 2, 3, 5)$.

Theorem: $S_n(A_1) + S_n(A_2) = S_n(A_1 \cdot A_2)$

where $A_1 \cdot A_2$ is the logical product of the classes i.e., the class of numbers which are common to A_1 and A_2. Thus $S_6(1, 2, 3) + S_6(2, 3, 5) = S_6(2, 3)$.

These theorems follow from the fact that a product is

zero if either factor is zero, while a sum is zero only if both terms are zero. The negative of a set of a-numbers will be written A' and means the class of all the numbers from 0 to n inclusive which are not members of A. Thus if A is the set of numbers 2, 3, and 5, and n = 6 then A' is the set of numbers 0, 1, 4, and 6.

Theorem: $S_n(A') = S_{\bar{n}}(A)$

These theorems are useful if several symmetric functions are to be obtained simultaneously.

Before we study the synthesis of a network for the general symmetric function consider the circuit a-b of Fig. 24. This circuit represents $S_3(2)$.

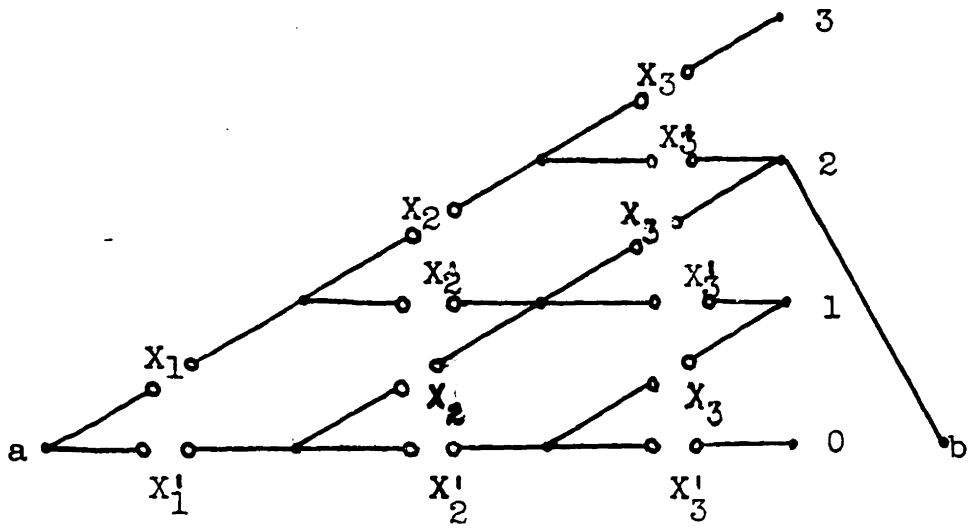


Fig. 24

The line coming in at a first encounters a pair of hinderances X_1 and X_1'. If $X_1 = 0$, the line is switched

up to the level marked 1, meaning that 1 of the variables is zero. If $X_1 = 1$, the line stays on the level marked 0; next hinderances X_2 and X_2' are encountered. If X_2 is zero, the line is switched up a level; if not, it stays at the same level. Finally reaching the right hand set of terminals the line has been switched up to a level representing the number of variables which are dqual to zero. Terminal b is connected to level 2 and therefore the circuit a-b will be completed if and only if 2 of the variables are zero. Thus the function $S_3(2)$ is represented. If $S_3(0,3)$ had been desired, terminal b would be connected to both levels 0 and 3. In figure 24 certain of the elements are evidently superfluous. The circuit may be simplified to the form of Fig. 25.

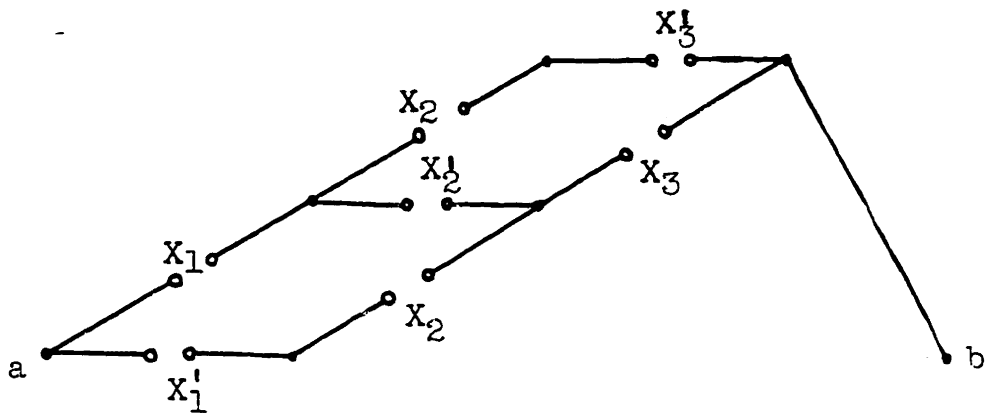


Fig. 25

For the general function exactly the same method is followed. Using the general circuit for n

variables of Fig. 26, the terminal b is connected to the levels corresponding to the a-numbers of the desired symmetric function. In Fig. 26 the hinderances are represented by simple lines, and the letters are omitted from the circuit, but the hinderance of each line may easily be seen by generalizing Fig. 24.

NOTE: All sloping lines have hinderance of the variable written below; horizontal lines have negative of this hinderance.

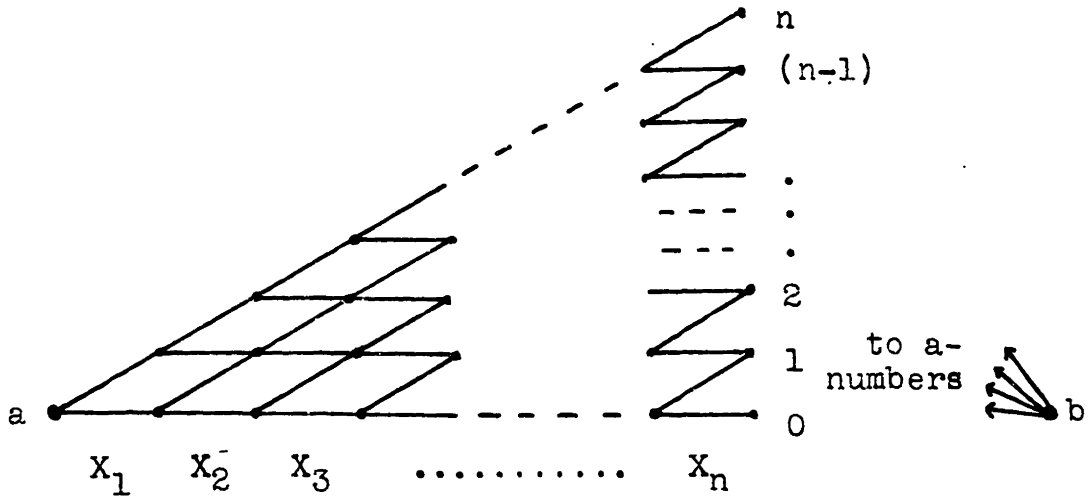


Fig. 26

After terminal b is connected, all superfluous elements may be deleted.

In certain cases it is possible to greatly simplify the circuit by shifting the levels down. Suppose the function $S_6(0,3,6)$ is desired. Instead of continuing the circuit up to the 6th level, we

connect the 2nd level back down to the zero level as shown in Fig. 27. The zero level then also becomes the 3rd level and the 6th level.

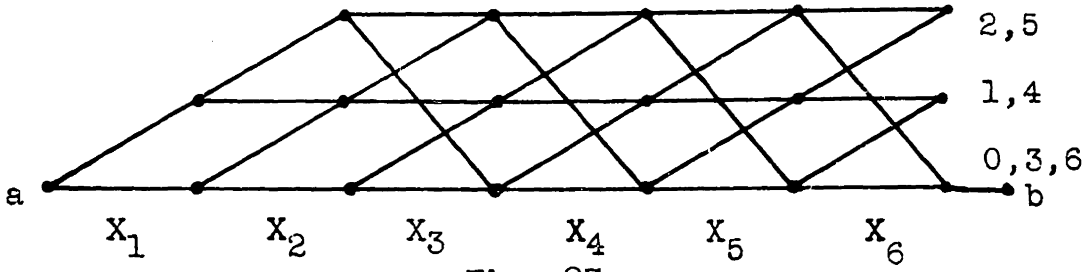


Fig. 27

With terminal b connected to this level, we have realized the function with a great saving of elements. Eliminating unnecessary elements the circuit of Fig. 28 is obtained. This device is especially useful if the a-numbers form an arithmetic progression, although it can sometimes be applied in other cases. The functions $\sum_1^n{}_2 X_k$ and $(\sum_1^n{}_2 X_k)'$ which were shown to require the most elements for a series-parallel realization have very simple circuits when developed in this manner. It can be easily shown that if n is even, then $\sum_1^n{}_2 X_k$ is the symmetric function with all the even numbers for a-numbers, if n is odd it has all the odd numbers for a-numbers. The function $(\sum_1^n{}_2 X_k)'$ is, of course, just the opposite. Using the shifting down process

the circuits are as shown in Fig. 29.

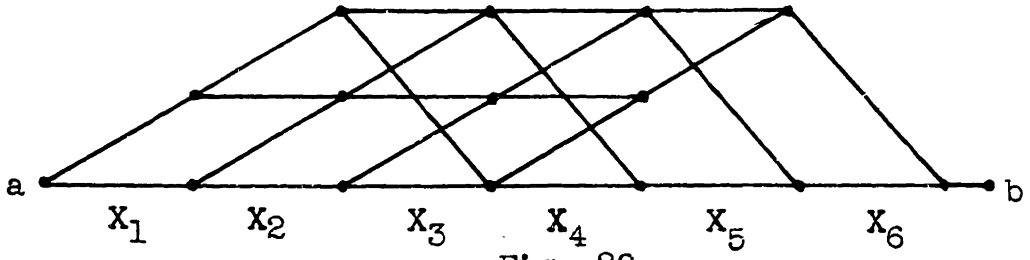
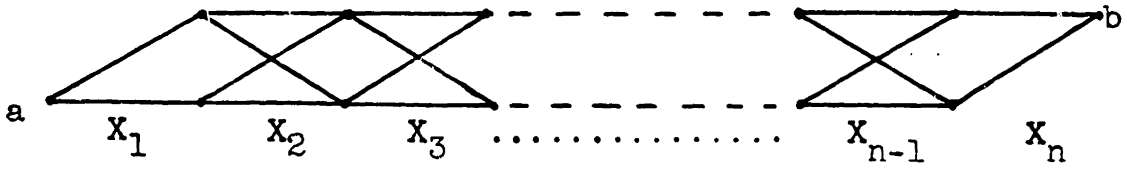
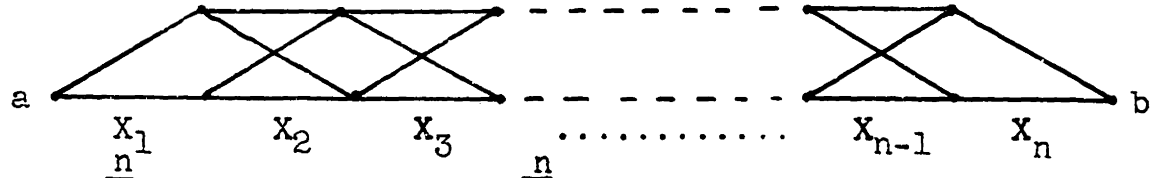


Fig. 28



$\sum_{1}^{n}{}_{2} X_k$ for n odd; $(\sum_{1}^{n}{}_{2} X_k)'$ for n even.



$\sum_{1}^{n}{}_{2} X_k$ for n even; $(\sum_{1}^{n}{}_{2} X_k)'$ for n odd.

Fig. 29

These circuits each require 4(n-1) elements. They will be recognized as the familiar circuit for controlling a light from n points: If at any one of the points the position of the switch is changed, the total number of variables which equals zero is changed by one, so that if the light is on, it will be turned off and if already off, it will be turned on.

The general network of Fig. 26 contains $n(n + 1)$ elements. It can be shown that for any given selection of a-numbers at least n of the elements will be superfluous. It follows that any symmetric function of n variables can be realized with at most n^2 elements.

Equations from Given Operating Characteristics. In general, there is a certain set of independent variables A, B, C,... which may be switches, externally operated or protective relays. There is also a set of dependent variables x, y, z.... which represent relays, motors or other devices to be controlled by the circuit. It is required to find a network which gives for each possible combination of values of the independent variables, the correct values for all the dependent variables. The following principles give the general method of solution.

1. Additional dependent variables must be introduced for each added phase of operation of a sequential system. Thus if it is desired to construct a system which operates in three steps, two additional variables must be introduced to represent the beginning of the last two steps. These additional variables may represent contacts on a stepping switch or relays which lock in sequentially. Similarly each required time delay will require a new variable, representing

a time delay relay of some sort. Other forms of relays which may be necessary will usually be obvious from the nature of the problem.

2. The hinderance equations for each of the dependent variables should now be written down. These functions may involve any of the variables, dependent or independent, including the variable whose function is being determined (as, for example, in a lock in circuit). The conditions may be either conditions for operation or for non-operation. Equations are written from operating characteristics according to Table II. To illustrate the use of this table suppose a relay A is to operate if x is operated and y or z is operated and x or w or z is not operated. The expression for A will be:

$$A = x + yz + x'w'z'$$

Lock in relay equations have already been discussed. It does not, of course, matter if the same conditions are put in the expression more than once--all superfluous material will disappear in the final simplification.

3. The expressions for the various dependent variables should next be simplified as much as possible by means of the theorems on manipulation of these quantities. Just how much this can be done depends somewhat

TABLE II

RELATION OF OPERATING CHARACTERISTICS AND EQUATIONS

Symbol	In Terms of Operation	In Terms of Non-Operation
X	The switch or relay X is operated.	The switch or relay X is not operated.
=	If.	If.
X'	The switch or relay X is not operated.	The switch or relay X is operated.
•	Or.	And.
+	And.	Or.
(--)'	The circuit (--) is not closed, or apply De Morgan's Theorem.	The circuit (--) is closed, or apply De Morgan's Theorem.

on the ingenuity of the designer.

4. The resulting circuit should now be drawn. Any necessary additions dictated by practical considerations such as current carrying ability, sequence of contact operation, etc., should be made.

V Illustrative Examples

In this section several problems will be solved with the methods which have been developed. The examples are intended more to show the versatility of relay and switching circuits and to illustrate the use of the calculus in actual problems than to describe practical devices.

It is possible to perform complex mathematical operations by means of relay circuits. Numbers may be represented by the positions of relays or stepping switches, and interconnections between sets of relays can be made to represent various mathematical operations. In fact, any operation that can be completely described to the required accuracy (if numerical) in a finite number of steps using the words "if," "or," "and," etc. (see Table II) can be done automatically with relays. The last two examples are illustrations of mathematical operations accomplished with relays.

A Selective Circuit

A relay A is to operate when any one, any three or when all four of the relays w, x, y, and z are operated. The hinderance function for A will evidently be:

$$A = wxyz + w'x'yz + w'xy'z + w'xyz' + wx'y'z + wx'yz' + wxy'z'$$

Reducing to the simplest series-parallel form:

$$A = w[x(yz + y'z') + x'(y'z + yz')] + w'[x(y'z + yz') + x'yz]$$

This circuit is shown in Fig. 30. It requires 20 elements.

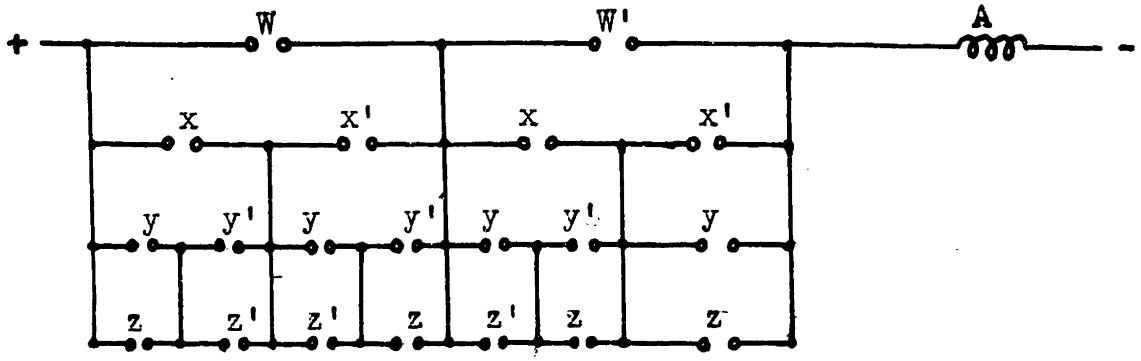


Fig. 30

However, using the symmetric function method, we may write for A:

$$A = S_4(1, 3, 4)$$

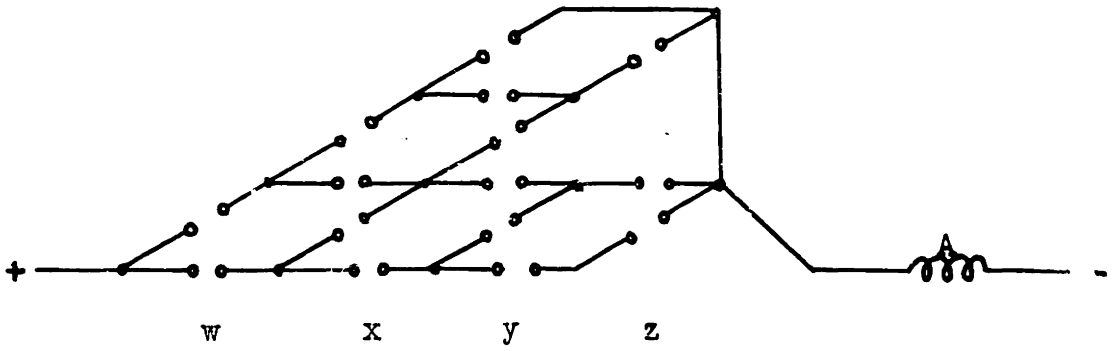


Fig. 31

This circuit contains only 15 elements. A still further reduction may be made with the following device. First write:

$$A' = S_4(0,2)$$

This has the circuit of Fig. 32. What is required is the negative of this function. This is a planar network and we may apply the theorem on the dual of a network, thus obtaining the circuit shown in Fig. 33.

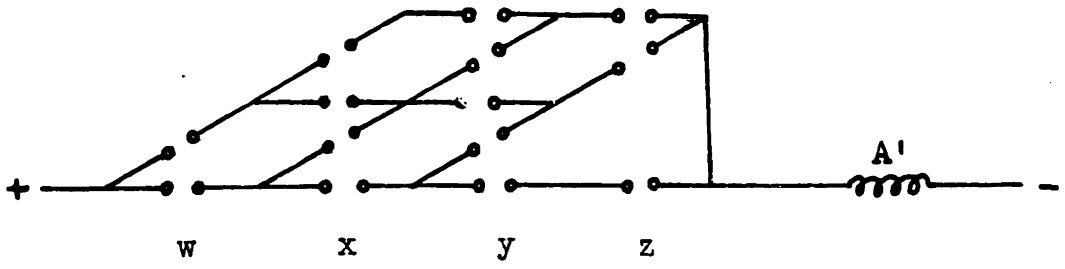


Fig. 32

This contains 14 elements and is probably the most economical circuit of any sort.

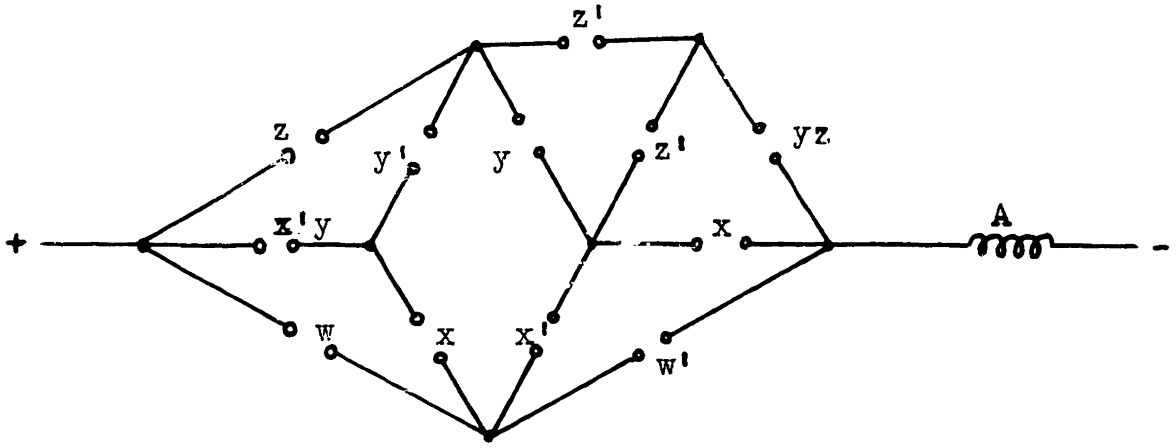


Fig. 33

Design of an Electric Combination Lock

An electric lock is to be constructed with the following characteristics. There are to be 5 push button switches available on the front of the lock. These will be labeled a, b, c, d, e. To operate the lock the buttons must be pressed in the following order - c, b, a and c simultaneously, d. When operated in this sequence the lock is to be unlocked, but if any button is pressed incorrectly an alarm U is to operate. To relock the system a switch g must be operated. To release the alarm once it has started, a switch h must be operated. This being a sequential system either a stepping switch or additional sequential relays are required. Using sequential relays let them be denoted by w, x, y, and z corresponding respectively to the correct sequence of operating the push buttons. An additional time delay relay is also required due to the third step in the operation. Obviously, even in correct operation a and c cannot be pressed at exactly the same time, but if only one is pressed and held down the alarm should operate. Therefore assume an auxiliary time delay relay y which will operate if either a or c alone is pressed at the end of step 2 and held down longer than time s the delay of the relay.

When z has operated the lock unlocks and at this point let all the other relays drop out of the circuit. The equations of the system may be written down immediately:

$$w = cw + z' + U'$$

misprint ? see circuit

$$x = bx + w' + z' + U'$$

$$y = (a + c)y + x + z' + U'$$

$$z = z(d + y) + g' + U'$$

$$v = x + y' + ac + a'c' + z' + U'$$

$$U = e(w' + abd)(w + x' + ad)(x + y' + dv(t-s))(y + b)U$$

$$U + h' + z'$$

These expressions can be simplified considerably, first by combining the second and third factors in the first term of U, and then by factoring out the common terms of the several functions. The final simplified form is as below:

$$U = \left| \begin{array}{l} \\ \\ \\ \\ \\ z' + \left| \begin{array}{l} \\ \\ \\ v' + \left| \begin{array}{l} cw \\ bx + w \\ x + \left| \begin{array}{l} (a+c)y \\ y' + ac + a'c' \end{array} \right. \end{array} \right. \end{array} \right. \end{array} \right. \quad h' + e[ad(b+w') + x'](x + y' + dv)(y + b)U$$

$$z = g' + (y + d)z + U'$$

This corresponds to the circuit on the following page.

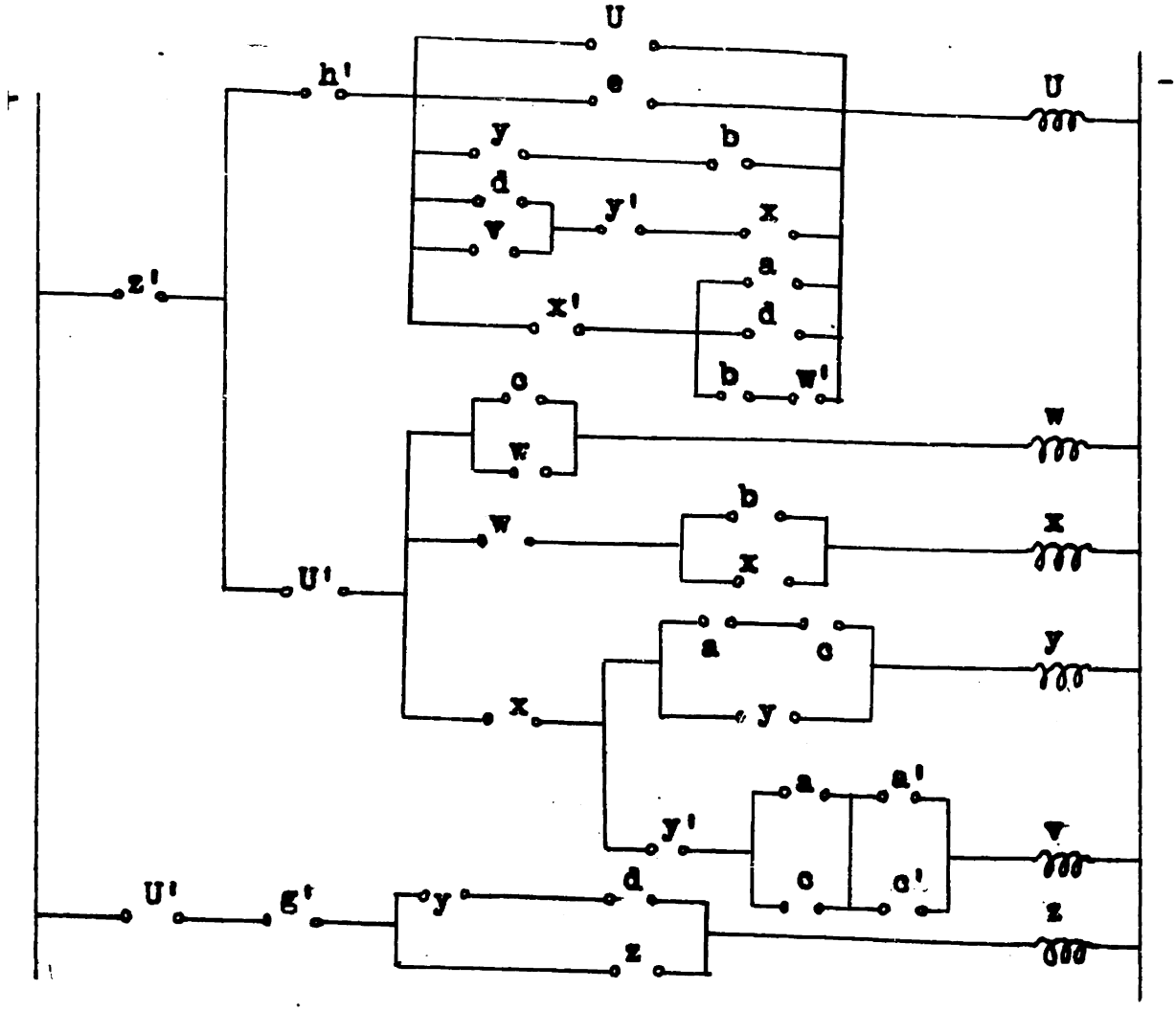


Fig. 34

A Vote Counting Circuit

A circuit is to be constructed with the following properties. There are to be thirteen lights, marked 0, 1, 2 ...12 and twelve two-position switches, x_1, x_2... x_{12}, one for each voter, each marked with two possible votes, yes or no. There is also a control button C. The lights are to count the number of 'yes' votes. If 5 voters move their switches to the 'yes' position and the remaining 7 vote 'no,' the light marked 5 is to light up providing the control button C is pressed, and similarly for any number of votes.

This is clearly an application of symmetric functions discussed previously. If we represent the lights by the symbols L_0, L_1, ... L_{12}, the the equations of the system will evidently be:

$$L_k = C + S_{12}(k) \qquad k = 0, 1, \ldots 12$$

The circuit representing this system according to the symmetric function development will be:

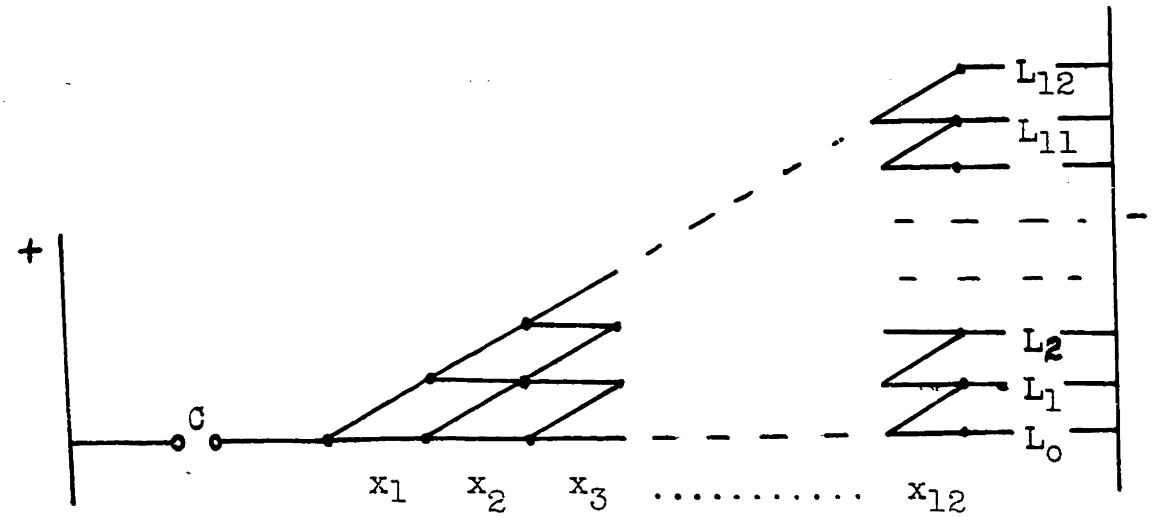


Fig. 35

Electric Adder to the Base Two

A circuit is to be designed that will automatically add two numbers, using only relays and switches. Although any numbering base could be used the circuit is greatly simplified by using the scale of two. Each digit is thus either 0 or 1; the number whose digits in order are a_k, a_{k-1}, a_{k-2}, ... a_2, a_1, a_0 has the value $\sum_{j=0}^{k} a_j 2^j$. Let the two numbers which are to be added be represented by a series of switches, a_k, a_{k-1}, ...a_1, a_0 representing the various digits of one of the numbers and b_k, b_{k-1}, ...b_1, b_0 the digits of the other number. The sum will be represented by the positions of a set of relays s_{k+1}, s_k, s_{k-1} ....s_1, s_0. A number which is carried to the jth column from the (j-1)th column will be represented by a relay c_j. If the value of any digit is zero, the corresponding relay or switch will be taken to be in the position of zero hinderance; if one, in the position where the hinderance is one. The actual addition is shown below:

c_{k+1}	c_k -----	$c_{j+1} c_j$ ----	c_2 c_1	carried numbers
	a_k -----	$a_{j+1} a_j$ ----	a_2 a_1 a_0	1st number
	b_k	$b_{j+1} b_j$	b_2 b_1 b_0	2nd number
c_{k+1} = s_{k+1}	s_k	s_j	s_2 s_1 s_0	Sum

starting from the right, s_0 is one if a_0 is one and b_0 is zero or if a_0 is zero and b_0 one but not otherwise. Hence:

$$s_0 = a_0 \, b_0' + a_0' \, b_0 = a_0 \oplus b_0$$

c_1 is one if both a_0 and b_0 are one but not otherwise.

$$c_1 = a_0 \, b_0$$

s_j is one if just one of a_j, b_j, c_j is one, or if all three are one.

$$s_j = S_3(1,3) \qquad \text{variables } [a_j, b_j, c_j]$$

c_{j+1} is one if two or if three of these variables are one.

$$c_{j+1} = S_3(2, 3) \qquad \text{variables } [a_j, b_j, c_j]$$

Using the method of symmetric functions, and shifting down for s_j gives the circuits of Fig. 36.

$j = 1, 2, \ldots k$ $\qquad\qquad$ $j = 0$

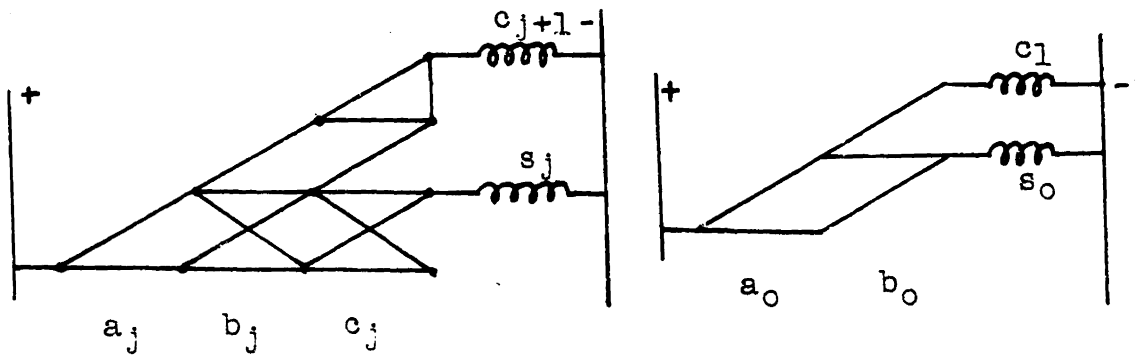


Fig. 36

Eliminating superfluous elements we arrive at Fig. 37.

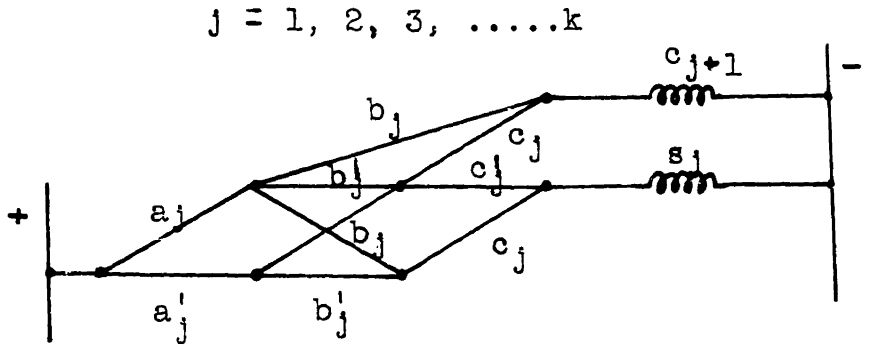


Fig. 37

A Factor Table Machine

A machine is to be designed which will automatically print a table of factors and primes of all the integers from 1 to 100,000,000. If a number is prime, it is to be so marked; if composite, its least factor is to be printed beside it. The principle which will be used is that of the sieve of Eratosthenes (6). Let the natural numbers be written in order:

1, 2, 3, 4, 5, 6, 7, 8,.....

Now consider the prime numbers in order, 2, 3, 5, 7, 11, 13, 17..... Each 2nd number after 2 in the row of natural numbers has the least prime factor 2; each third number after 3 which is not a multiple of 2 has the least prime factor 3; each 5th number after 5 not divisible by 2 or 3 has the least prime factor 5, etc. Any number P not having a prime factor less than itself is, of course, a prime. It is customary in tables of this sort to omit numbers divisible by 2, 3, or 5 thus reducing the number of integers which need be considered to 4/15 of the largest number N (10^8 in this case). It should also be noted that any composite number less than or equal to N has a least factor less than or equal to $\sqrt{N}$. Thus in our case only primes less than 10,000 need be considered in the filtering

process described. The asymptotic formula N/ln N (for the number of primes less than N) shows that there are about 1000 primes less than 10,000. Let each of these primes after 5 be represented by a counter C_k with the following properties. There are three magnets, M_2, M_4, and M_6. When M_2 operates all the counters are advanced 2 units; M_4 and M_6 advance the counters 4 and 6 units respectively. The purpose of these magnets is to automatically omit numbers divisible by 2, 3, and 5. Note that starting with 1 the next number not divisible by 2, 3, or 5 is 7, an advance of 6; the next advance is 4 (to 11), then 2 (to 13). The total cycle of advances is as follows:

6, 4, 2, 4, 2, 4, 6, 2 (1)

after which the same series is repeated (the period is 30, the least common multiple of 2, 3, and 5). As the successive numbers are considered for factors or primality, the counters will advance according to this sequence. When any counter C_k representing the prime P_k reaches the value of this prime it is to be so constructed that it automatically makes a connection X_k. Each counter is to have a return magnet R_k, which when activated returns the counter to zero. The general operation of the device will then be as follows. Starting at the number 1 (the counter and printer

representing the number being considered set at 1) and with the counters representing the primes less than 10,000 all set at zero, the counters are advanced according to the sequence (1). If for any number N, X_k makes contact, then P_k is a factor of N; the least P_k being the least factor. If no X_k makes contact, N is a prime. When any X_k makes contact, it is to be automatically returned to zero by means of R_k. To record the results a printer U_k should be associated with each counter which will print the value of the prime P_k opposite N when magnet U_k is activated. If N is a prime, a printer S should print a symbol to call attention to the fact.

Although this entire design could be carried out with relays alone, it is probably more economical to construct the counters on mechanical principles, and therefore only the control circuits will be described. To automatically advance the numbers at short intervals some kind of an impulse generator is necessary. The simplest method of obtaining this is to use a relay with a small time delay δ. If the relay is labeled Z(t), then the contacts have a hinderance function $Z(t-\delta)$, and the connection $Z(t) = Z'(t-\delta)$ will give a series of impulses of period 2δ. The sequence of advances may be easily obtained with an 8

point rotary switch. Let this switch have a magnet L which advances the switch one point when activated. Then if we connect L so that $L = Z(t-\delta)$ and connect the contacts of the rotary switch to the magnets M_2, M_3, and M_5 according to the order of (1), the counters will all be advanced periodically in this sequence. After the counters have advanced a step, certain of the X_ks will equal zero if the number is composite. In this case these X_ks should cause the smallest factor to print and then return to zero. This condition will be satisfied by the following equations:

$$U_k = X_k + \sum_{j=1}^{k-1} X'_j + z'(t-\delta) \qquad k = 1, 2, 3.....$$

$$R_k = X_k + y(t-e) \qquad (2)$$

$$y(t) = z(t-\delta)$$

That is, the printer U_k operates if $X_k = 0$ and the X_js, $j<k$, do not equal zero. Also after a delay e to allow for printing, the counter is returned to zero. If none of the X_ks make contact on a number N, it is a prime and S should print. This can be accomplished with the following equation:

$$S = z(t-\delta) + \sum X'_j$$

The main printer and counter N should print on each number.

$$N = s'(t-\delta)$$

Using the method of factoring of simultaneous equations the system (2) can be greatly simplified as follows:

$$\begin{aligned} U_1 &= X_1 \\ U_2 &= X_2 \\ U_3 &= X_3 \\ &\\ U_n &= X_n \\ S &= \end{aligned} \Big| + X'_1 \Big| \quad + X'_2 \Big| \cdots \Big| + X'_{n-1} \Big| \quad + X'_n$$

The circuit of the entire device is shown schematically in Fig. 38.

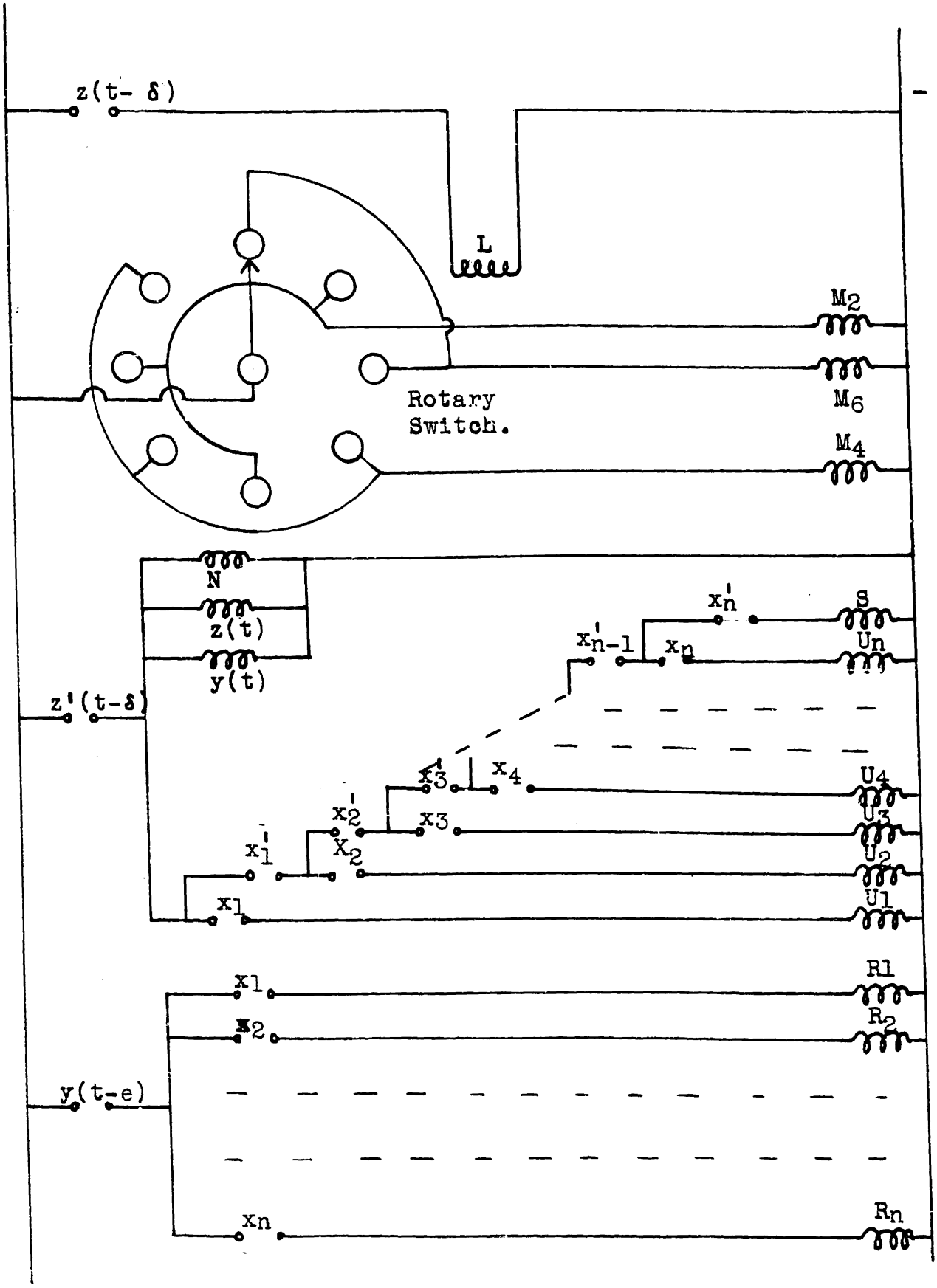

z(t- δ)
L
M2
M6
M4
Rotary
Switch.
N
z(t)
y(t)
z'(t-δ)
x'n
S
x'n-1
xn
Un
x'3
x4
U4
x'2
x3
U3
x'1
X2
U2
x1
U1
x1
R1
x2
R2
y(t-e)
xn
Rn

Fig. 38

This design requires that the primes less than 10,000 be known. If desired, the machine could be made to automatically connect in new counters as the primes were found, but there are many accurate tables of primes up to 10,000 so that this would not be necessary.

As to the practicability of such a device, it might be said that J.P. Kulik spent 20 years in constructing a table of primes up to 100,000,000 and when finished it was found to contain so many errors that it was not worth publishing. The machine described here could probably be made to handle 5 numbers per second so that the table would require only about 2 months to construct.

REFERENCES

(1) A complete bibliography of the literature of symbolic logic is given in the Journal of Symbolic Logic, Vol. 1, No. 4, December 1936. Those elementary parts of the theory that are useful in connection with relay circuits are well treated in the two following references.

(2) Louis Cauturat, The Algebra of Logic, The Open Court Publishing Co.

(3) A. N. Whitehead, Universal Algebra, Cambridge, at the University Press, Vol. I, Book II, Chapters I and II, pp. 35-82.

(4) E. V. Huntington, Trans. of the American Mathematical Society, Vol. 35, 1933, pp. 274-304. The postulates referred to are the fourth set, given on page 280.

(5) George Boole, Finite Differences, G. E. Strechert & Co., Chap. X.

(6) L. E. Dickson, History of the Theory of Numbers, Vol. I, Carnegie Institution of Washington, Chap. XIII.