REPORT R-414 MAY, 1969

CSL COORDINATED SCIENCE LABORATORY

GENERATION OF DIAGNOSTIC TESTS USING PRIME IMPLICANTS

MICHAEL REID PAIGE

UNIVERSITY OF ILLINOIS - URBANA, ILLINOIS

GENERATION OF DIAGNOSTIC TESTS USING PRIME IMPLICANTS

Michael Reid Page

Department of Electrical Engineering
University of Illinois May 1969

This work was supported in part by the Joint Services Electronics Program (U.S. Army, U.S. Navy, and U.S. Air Force) under Contract DAAB 07-67-C-0199; and in part by National Science Foundation GK-1663.

SECTION 1: INTRODUCTION

With recent advances in microminiaturization and fabrication techniques, digital computer design has begun to make fuller use of single packages containing many logic gates. This change in design approach has shifted the emphasis in fault testing techniques. Fault Testing is defined here as the process of determining whether or not a package of gates is operating correctly by comparing its output under certain inputs with that of the fault-free machine.

It is necessary to distinguish two terms, fault and error. In this paper the word "fault" will signify a malfunction of an individual line, while "error" will be the discernable result of faults on the output function.

The problem of testing circuits can be limited to fault detection, that is, to checking whether or not any faults are present. The methods employed toward this end may involve such concepts as path sensitizing,[1] which is a technique by which the output is made dependent on a

---

[1] Armstrong, D. B., "On Finding a Nearly Minimal Set of Fault Detection Tests for Combinational Logic Nets," IEEE Transactions on Electronic Computers, Vol. EC-15, No. 1, pp. 66-73 (February 1966).

LIST OF FIGURES

TABLE OF CONTENTS

SECTION Page

ACKNOWLEDGEMENT

The author is deeply indebted to his advisor, Dr. Gernot Metze, for his patient guidance and inspired insight. Dr. Metze's availability for comment and his wealth of ideas have greatly aided in the formulation of this paper.

A note of thanks is also due to the members of the Switching Systems Group of the Coordinated Science Laboratory for their interchange of ideas. The author wishes to particularly extent his appreciation to Messers. John Hayes and Donald Schertz whose timely comments made the research and writing of this thesis easier.

A final note of thanks is given to Miss Natalie Knowles for her patience and understanding.

particular line by deliberate selection of input parameters. Another procedure which may be used is to exhaust all possible input combinations.

It is the purpose of this thesis to present an approach to fault testing using only the prime implicant information provided for the system. This technique is an answer to the question, "How may fault tests for a given circuit be selected with the minimum of detailed analysis of the logic network?" The number of tests specified by this approach is, in general, much lower than the number of possible input combinations, although it may be higher than the minimum number of tests which could be derived with full, detailed knowledge of the circuit structure.

In Section 2, the selection of fault tests for single gates (AND, OR, NAND, and NOR) is unified into a general procedure. This technique considers all logical changes that the output function of the gate can suffer due to any of the line faults. Using these output changes, all line faults can be considered as divided into two classes. The first class includes all faults which cause the output function to become a constant (either 1 or 0). The second class contains all faults not in the first class, and thus can be described as the class of all faults which cause a logical change in the output function, but do not make it a constant.

These two fault classes are useful to classify fault tests as either existence tests or growth tests, according to whether they test faults in the first or the second of the above classes. The significance of the names for the tests and the method for selection of these tests is presented in the thesis.

Section 3 contains three individual points of discussion, the results of which are germane to the proof presented in Section 4.

The extension of growth and existence tests to two 2-level irredundant networks, where one level consists of AND gates and the other of ORs, is made in Section 3.1. A detailed analysis of these two configurations of gates yields the information that, in each case, all faults in the entire system will be checked if the above two types of tests are applied solely to the second level of gates (levels are numbered from the output backwards).

A discussion about the significance of choosing the proper form for the representation of the output function is contained in Section 3.2. For this technique of fault testing, the output function must reflect the internal structure of the network's logic.

Section 3.3 contains a discussion on the topic of selecting a set of tests that will be valid for any irredundant disjunctive form of a function that has several irredundant representations. A procedure is presented that allows such a set of tests to be chosen.

The preceding discussions are tied together in a theorem presented in Section 4 which states, briefly, that the growth and existence tests developed for a two-level irredundant implementation of a network will be sufficient to test for all faults in the equivalent n-level network. This theorem is followed by several corollaries extending it to networks with all NANDS or all NORS.

SECTION 2: FAULT TESTING SIMPLE GATES

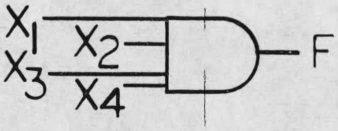
The purpose of this Section is to describe the operation of gates and to generalize the errors due to individual line faults. This analysis will lead to the definition of growth and existence testing of gates. We will assume that only single faults occur, and that they may be represented by lines "stuck-at-zero" or "stuck-at-one."

The output function of an AND gate, shown in Figure 1a, can change its value in one of two ways. The fault-free output of this gate is written as

$$F = X_1X_2X_3X_4$$

where X_i represents the input variable on the ith line. If any input line is stuck-at-zero, then $F = 0$. If, on the other hand, any input line is stuck-at-one, then F becomes independent of the correpsonding input variable.

In the Karnaugh Map representation of this gate, as shown in Figure 1b, the above mentioned errors become more apparent. If $F = 0$, the original minterm of F is replaced by 0. If F becomes independent of any variable, the function changes to include minterms that indicate this condition.



1a. Four-Input AND Gate

	X_1X_2			
X_3X_4	0	0	0	0
	0	0	0	0
	0	0	1	0
	0	0	0	0

1b. F For Fault-free Gate

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

1c. F = 0

0	0	0	0
0	0	1	0
0	0	1	0
0	0	0	0

1d. F Independent of X_3

Figure 1

A Four-Input AND Gate and
Associated Karnaugh Maps

Using the above approach, the OR gate may be shown to have similar properties when its maxterms are considered. For the OR gate shown in Figure 2a, the output function is written as

$$F = X_1+X_2+X_3+X_4.$$

If any input line is stuck-at-one, then F = 1. If, however, any input line is stuck-at-zero, then F becomes independent of the corresponding input variable.

The Karnaugh Map for this gate, as in Figure 2b, displays the above mentioned errors graphically. If F = 1, the maxterm of the original function is replaced by 1. If the function becomes independent of any variable, the function changes to include maxterms that indicate this condition.

NOR and NAND gates, because of their similarity to AND and OR, respectively, will not be discussed in detail. However, it is evident that the same type of investigation on the effects of stuck-at-one and stuck-at-zero line faults will yield similar results to those mentioned for the AND and OR gates.

Before presenting a procedure for fault testing these gates (OR,AND,NOR, and NAND) several terms will be defined.

Definition 1: The growth of a product function consists of those minterms which are ORed to the prime implicant of the AND gate when the function becomes independent of any of its input variables. Similarly, the <u>growth</u> of a sum function consists of those maxterms which are ANDed to the prime implicant of the OR gate when the function becomes independent of any of its input variables.

<u>Definition 2</u>: A <u>growth test</u> is a test to check if a given prime implicant (implicate) has changed to become independent of any one variable.

<u>Definition 3</u>: An <u>existence test</u> is a test to establish whether or not a gate has suffered a fault which changes the output function to 0(1).

Using the above terms the selection of fault detection tests may proceed as follows:

(1) At least one existence test must be used for the gate in question. This is done by selecting any single minterm (maxterm) from the prime implicant (implicate) expression for the gate.

(2) At least one growth test must be made for each of the line variables of the function. This is done by selecting one of the minterms(maxterms) which belong to the growth due to that variable.

The set of tests thus selected covers all possible stuck-at-zero or stuck-at-one faults that may occur. It is important to note that faults on the output line of a

2a. Four-Input OR Gate

	X_1X_2			
X_3X_4	0	1	1	1
	1	1	1	1
	1	1	1	1
	1	1	1	1

2b. F For Fault-free Gate

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

2c. F = 1

0	1	1	1
1	1	1	1
1	1	1	1
0	1	1	1

2d. F Independent of X_3

Figure 2

An OR Gate and Associated Karnaugh Maps

gate are not individually analyzed. A stuck-at-one fault on an output line causes F = 1, which for an AND gate is caught by any of the growth tests which check whether any minterms were added to the original function. For an OR gate, F = 1 implies that all maxterms have been replaced by cones, and this condition will be checked by the existence test. In a similar manner, the output line stuck-at-zero which forces F = 0 is checked by the existence test for the AND and any growth test for the OR gate.

It is obvious that the indicated tests would detect the opposite output faults in the case of NANDs and NORs. If we consider the NOR gate in Figure 3, then the following fault detection tests can be selected:

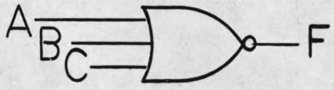
(0,0,0) as the existence test

(0,0,1),(0,1,0), and (1,0,0) as the growth tests, or more compactly, in decimal notation,

0 as the existence test

1,2,4 as the growth tests.

It is important to note that information about the prime implicant of a gate is all that is necessary to develop a minimal set of fault tests. This minimal set results if only one existence test and only one test for each of the possible growths is selected.



3a. Network

AB

C			
^{0} 0	^{2} 1	1	^{4} 1
^{1} 1	1	1	1

3b. Fault-free Map

0	1	1	0
1	1	1	1

3c. F Independent of A

0	0	1	1
1	1	1	1

3d. F Independent of B

0	1	1	1
0	1	1	1

3e. F Independent of C

Figure 3

A NOR Gate and Associated Karnaugh Maps

SECTION 3: TWO-LEVEL IRREDUNDANT IMPLEMENTATIONS

3.1. Fault Tests for Irredundant Two-Level Realizations

In this section the topic of growth and existence testing will be extended to irredundant 2-level implementations. The possibility of interchanging the testing of minterms and maxterms of a given function will be discussed. Finally, the selection of a set of tests which is valid for any implementation of a function with several distinct irredundant disjunctive forms is presented.

In a 2-level AND-OR irredundant realization, as shown in Figure 4, the output may be written as

$$F = X_1X_2 \ldots\ldots\ldots X_n \ + \ \ldots\ldots\ldots \ + \ \ldots\ldots\ldots$$

Each product term of F is responsible for at least one unique minterm, since the function is irredundant.

We can determine the contribution of any one AND gate to the final output, since it must be responsible for one of the prime implicants of F. If we isolate this gate we can predict, using the methods of Section 2, the errors it contributes. In this manner each AND gate may be analyzed.

Faults on lines of the OR gate can be shown to contribute no new information to our analysis. An OR input stuck-at-zero would normally be tested by checking if the output

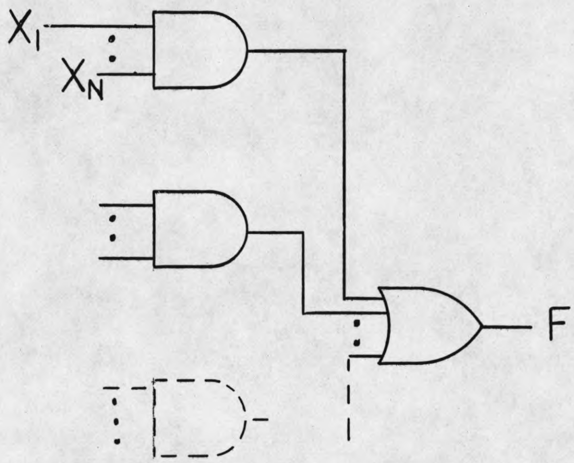
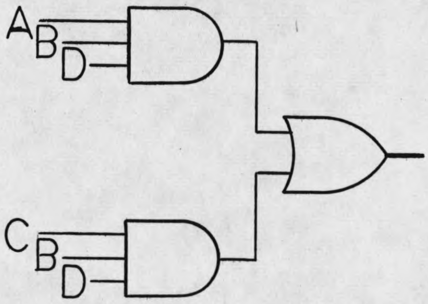


Figure 4
Two-Level AND-OR Network

has become independent of that variable. If we select an existence test for the AND gate corresponding to that line, we will then check for the stuck-at-zero fault at the AND output, which, of course, is indistinguishable from the OR input stuck-at-one. Similarly, an OR input stuck-at-one forces the output of the OR to one, which will be covered by the growth test of any of the individual AND inputs to the OR. Analysis of the OR output line stuck-at-one yields the same information as an input line with the same fault. If the output line is stuck-at-zero, then F = 0, which implies that all prime implicates are zero. This situation is checked by the existence tests for any of the AND gates.

Selection of fault tests for this two 2-level implementation may be made using the growth and existence tests of the individual AND gates. However, there is no need to allot one growth test per gate per fault since we do not care which gate has a fault, but, rather, whether or not a fault exists. Therefore, growth tests should be placed in areas of maximum overlap of the growths of the different gates so that several growths may be checked with one test. However, no minterm cell can be considered as a suitable growth test if it is included in the prime implicant of the network, since that cell will always be 1.



	AB			
CD	0	0	12 0	0
	0	5 0	13 1	9 0
	3 0	7 1	15 1	11 0
	0	0	14 0	0

Figure 5

A Two-Level AND-OR Network and

Associated Karnaugh Map

The selection of mutual growth tests is shown in Figure 5 for the network which realizes

$$F = ABD + BCD$$

ABD, which is realized as (13,15) can grow into (5),[2] (9,11), and (12,14). BCD,(7,15), can grow into (5),[3] (3,11),(6,14). In this example, the set of growth tests (5),(11), and (14) will check all possible growths.

Two-level irredundant OR-AND implementations can be tested using the concepts of fault testing for the individual OR gates, with similar results. Two-level implementations using NANDs or NORs can be tested by replacing them with equivalent AND-OR or OR-AND realizations, respectively.

It is important to note that the above procedure develops a minimal set of fault tests for any of the two-level implementations considered. This result emphasizes the important feature of this approach, that is, the minimal amount of detailed analysis necessary.

---

[2] (7) is not included as a growth of ABD since it is a minterm of BCD, and therefore already equal to 1.

[3] (13) is not included as a growth of BCD since it is a minterm of ABD.

3.2. Significance of the Form of the Output Function

The use of information concerning the implementation of a given function becomes more explicit if the testing of minterms and maxterms is interchanged. We may express any function in either a sum of products or a product of sums form, denoted S of P or P of S respectively. We will define F_s as the S of P form and F_p as the P of S form. The minterms of F_s can only grow into those positions occupied by the maxterms of F_p, and vice versa. The growth of a product of F_s results in a prime implicant of F_p being reduced to 1, and likewise, the growth of a factor of F_p forces a factor of F_s to be reduced to 0. The testing of minterms, however, does not yield the same results as testing for maxterms of the same function, because the minterms of F_s do not necessarily grow to replace all the maxterms of F_p.

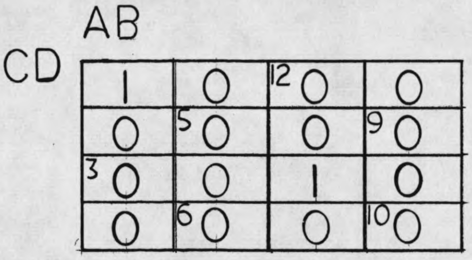
This conclusion is emphasized in the following example. Consider the function

$$F_s = ABCD + \bar{A}\bar{B}\bar{C}\bar{D}$$

and its alternate form

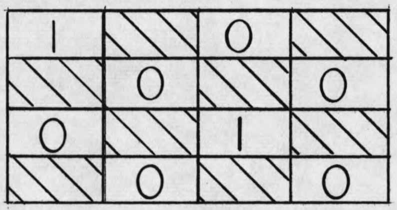
$$F_p = (A+\bar{D})(\bar{C}+D)(\bar{A}+B)(\bar{B}+C)$$

as shown in Figure 6. If we allow the minterms of F_s to grow into all possible positions, as shown by the shaded area in the Karnaugh Map, then the following maxterms are



1	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0

6a. Normal Function



1	▨	0	▨
▨	0	▨	0
0	▨	1	▨
▨	0	▨	0

6b. Growth of Minterms

Figure 6

Karnaugh Maps for $F = ABCD + \bar{A}\bar{B}\bar{C}\bar{D}$

left untouched: (3,5,6,8,9,10,12). If the maxterms are the basis for fault testing, then any of the above maxterms can be selected as an existence test. Many of the maxterms that are included in the growth of the minterms may then be omitted as tests.

The preceding discussion implies that the form of the output function must be known for this procedure of fault testing to work. It is a salient point that the structure of the network is reflected in the form of the output function; that is, for example, if the network is of AND-OR form, then the output function is a sum of products.

3.3. Functions With Several Distinct Irredundant Disjunctive Forms

If a given function has several distinct irredundant disjunctive forms, a set of growth and existence tests can be selected which is applicable for any of these forms. The procedure for selection of existence tests for such a function is as follows:

E1: List all possible prime implicants in their decimal form. Select one minterm at random as the first test.

E2: Remove from the list the minterm selected in step E1, and all prime implicants in which it appears.

Also delete from the listing all minterms which appear in any of the prime implicants deleted.

E3: From the remaining prime implicants select existence tests for those prime implicants which have had any of their minterms removed.

E4: Go back to step E2 unless all prime implicants are deleted from the list.

The above procedure for the selection of existence tests simply selects one test which serves as an existence test for several prime implicants. All prime implicants which share any minterms with the prime implicants tested by the first choice are next tested. This process continues until all prime implicants are exhausted.

Consider the function

$$F = \bar{A}CD\bar{E} + ABCD + \bar{B}CDE$$

which may alternately be written

$$F = BCD\bar{E} + \bar{A}\bar{B}CD + ACDE$$

as shown in Figure 7. Then, for this example, the set of existence tests which will be valid for either form can be selected as follows:

E1: The prime implicants are (3,7)(7,15)(3,19)(15,31)(19,27)(31,27), from which we select (3) as the first test.

E = 0

	AB			
CD	0	0	0	0
	^{1}0	^{5}0	^{13}0	^{9}0
	^{3}1	^{7}1	^{15}1	^{11}0
	^{2}0	^{6}0	^{14}0	^{10}0

E = 1

0	0	0	0
^{17}0	^{21}0	^{29}0	^{25}0
^{19}1	^{23}0	^{31}1	^{27}1
^{18}0	^{22}0	^{30}0	^{26}0

Figure 7

Karnaugh Map for $F = \bar{A}CD\bar{E} + ABCD + \bar{B}CDE$

$(F = \bar{A}\bar{B}CD + BCDE + BCDE)$

E2: After deleting all terms implied by step E1, the list is reduced to (15)(27)(15,31)(27,31).

E3: Tests (15) and (27) are next selected.

E4: Complete deletion of all remaining prime implicants is accomplished in this step so the procedure is completed. The final result (3,15,27) is the desired test set.

Growth testing proceeds in a similar manner to existence testing; this may be stated as:

G1: List all minterm growths of all prime implicants. Delete all growths which may be included in other growths or in other prime implicants.

G2: All sets with single elements <u>must</u> be tested for. Select all such tests.

G3: Select from all minterms a test at random.

G4: Remove all growths in which the test in step G3 occurs. Also remove from all remaining growths any minterms that may appear in any of the growths previously deleted.

G5: Perform step G2 if applicable.

G6: Select growth tests from any of the growths which have had any minterms removed due to the above process.

G7: Go back to step G4 if a growth remains contested; if at any time G6 cannot be applied to the remaining growths, then begin again with step G3. The process may be reinitiated at G3 as many times as necessary until all growths are deleted from the listing.

The above procedure may be described in the following manner. Growths with single elements which are not included in other sets must be tested individually since they represent areas that would not be tested otherwise. Selection of the first test checks several growth areas. All growths which share tested minterms growths are then tested. This process is continued, with restarts if necessary, until all growths are tested.

Returning to the previously introduced example, Figure 7, we can select growth tests which will be valid for any one of the forms the output may take. The results would be as follows:

G1: The growths for all prime implicants are

for (3,7) : (1,5)(11)(2,6)(23)
for (7,15) : (11)(23)(5,13)(6,14)
for (3,19) : (11)(23)(2,28)(1,17)
for (15,31) : (11)(23)(30,14)(13,29)
for (19,27) : (11)(23)(17,25)(18,26)
for (31,27) : (11)(23)(30,26)(29,25)

This collection can be reduced to (11)(23)(1,5)(2,6)(5,13)(6,14)(1,17)(2,18)(13,29)(14,30)(17,25)(18,26)(29,25)(30,26).

G2: The tests (11) and (23) must be made.

G3: Select (1) as a test.

G4 to G6: Completing theprocess using the test from step G3; we will select tests (25) and (13).

G7: At this point the original list is reduced to (2,6)(6,14)(2,18)(14,30)(18,26)(30,26) and a choice similar to that in step G3 is indicated. We will choose (2) as the test, and following the procedure as outlined tests (14) and (26) are also chosen.

The final set of growth tests for this example is (1,2,13,14,25,26,11,23).

SECTION 4: IRREDUNDANT N-LEVEL REALIZATIONS

The purpose of this section is to extend the concepts of growth and existence tests to irredundant N-level realizations with certain structures. A proof of the validity of the use of growth and existence tests for these structures is presented.

Definition 4: An (F,Y) stage is a two-level logic network where the first level consists of Y type gates and the second of F type gates.

It is noted here that levels and stages are counted from the primary output backwards into the system. An AND-OR 2-level realization is, thus, an (AND,OR) stage using the above definition.

Theorem: Faults in the nth (AND,OR) stage of an irredundant k (AND,OR) stage network can be tested for at the output using growth and existence tests.

Proof: Let F'' be the output of an AND gate in the nth (AND,OR) stage of a k (AND,OR) stage network. Assume that $X_1, X_2, \ldots\ldots, X_m$ are inputs to this particular AND gate.

Hence,

$$F'' = X_1 X_2 \ldots\ldots\ldots X_m$$

If we consider the path traced from this AND gate to each of the AND gates in the first (AND,OR) stage where it is a factor, we can represent each of these paths schematically as shown in Figure 8. Each OR gate in this path performs a disjunctive operation on F", thus the expression for F" is not changed by these gates. F" is changed, however, by the AND gates that it passes through.

Note that in Figure 8 all inputs to each gate shown, except for that one due to the path of F", are made a single vector input $Y_{s,t}$. The first subscript of this notation represents the gate using "a" for AND and "o" for OR. The second subscript gives the number of the (AND,OR) stage the gate is in.

The output of the AND gate in the first (AND,OR) stage may be written as

$$F' = (F'')\,(Y_{a,n-1}Y_{a,n-2}\cdots\cdots Y_{a,2}Y_{a,1})$$

If the output function of this network is realized in a 2-level AND-OR form, i.e., one (AND,OR) stage, then F' will be the output of one of the AND gates. This is true because of the irredundancy of the network.

If we perform an existence test on F' we will also be performing an existence test on F". This is true because the existence test for F' will check if F' = 0, and from the above equation F" = 0 if F' = 0. It then follows that the existence test for F' also checks the existence of F".

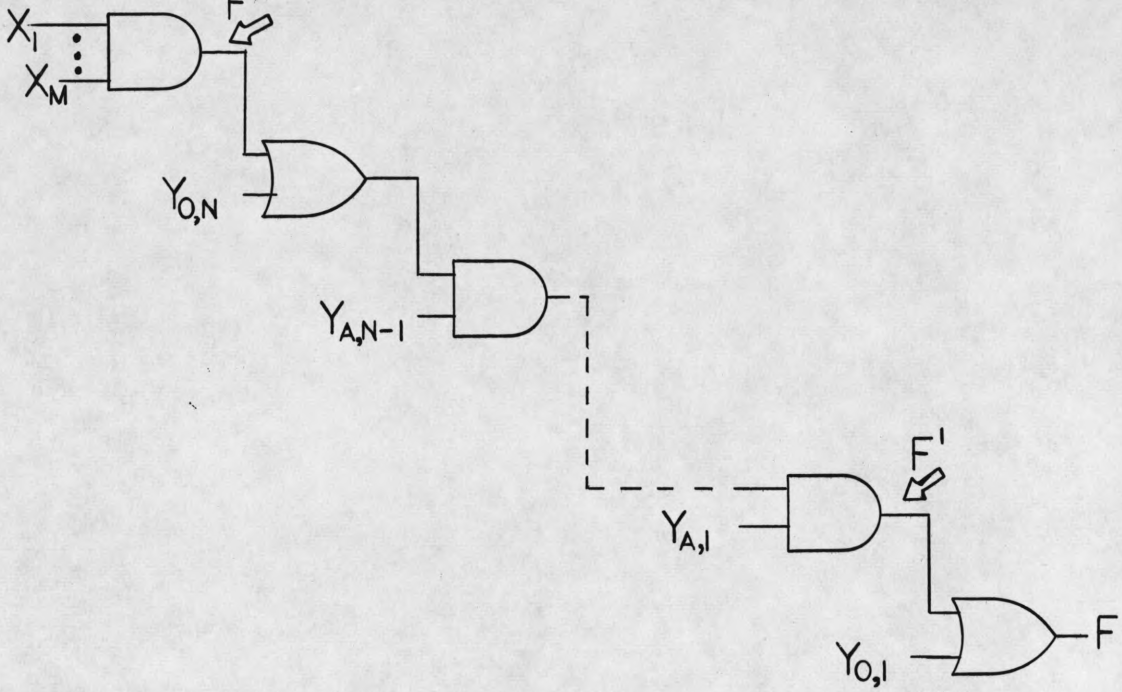


Figure 8

Path Traced from AND Gate to Output

A growth test for F', by definition, checks for all changes in F' due to its independence of any of its variables. Since F' may be written as

$$F' = (X_1 X_2 \ldots\ldots\ldots X_m)(Y_{a,n-1} Y_{a,n-2} \ldots\ldots\ldots Y_{a,2} Y_{a,1}),$$

a growth test for F' must check for its independence from any of the X_i variables of F".

Each of the input variables of F" may, in turn, be a function of several variables. In a similar manner to the above method of investigation, the existence and growth tests for F' can be shown to check for faults in these variables.

It has been previously shown (Section 2) that an OR gate following several ANDs does not have to be individually fault tested if each AND is tested. Therefore, by replacing this k (AND,OR) stage network by its irredundant one (AND,OR) stage equivalent,[4] all faults at <u>all</u> gates can be tested for by applying existence and growth tests to the output function.

QED

The following Corollaries are presented without proof.

---

[4] In Section 3.3 a method for selection of a set of tests that is valid for any form that a function with several irredundant disjunctive forms is presented. Using this information, it can be seen that any irredundant of these 2-level equivalents may be chosen.

Corollary 1: Faults in the nth (OR,AND) stage of an m (OR,AND) stage network can be tested for at the output using growth and existence tests.

Corollary 2: Faults in the nth level of a k-level NAND network can be tested for at the output using growth and existence tests.

Corollary 3: Faults in the nth level of a k level NOR network can be tested for at the output using growth and existence tests.

It is important to note that all of the above cases require the network to be irredundant.

The use of the knowledge that the form that the output function takes is a salient point in the fault testing of the network, becomes very apparent in the n(F,Y) stage case. We have shown that unnecessary tests may be selected by choosing the wrong form for the output and that some necessary tests may be omitted. It is important then that the 2-level equivalent representation be of the same form as the network.

Although the set of tests selected is not necessarily minimal, the amount of analysis needed to select these tests is quite small compared to other techniques available.[5,6,7]

---

[5] Armstrong, op. cit.

[6] Poage, J. F., "Derivation of Optimum Tests to Detect Faults in Combinational Circuits," Proceedings of the Symposium on Mathematical Theory of Automata, pp. 483-528 (Polytechnic Press, New York, N. Y., April 1963).

[7] Roth, P. J., "Diagnosis of Automata Failures: A Calculus and a Method," IBM Journal of Research and Development, Vol. 10, No. 4, pp. 278-291 (May 1962).

SECTION 5: CONCLUSIONS

"What is the minimal amount of information needed to develop a reasonable set of fault tests for a given network?" is the question that prompted the writing of this paper. Prime implicant information is shown to be the important key to the answer. Through the use of such concepts as growth and existence testing near minimal sets of fault tests are derived with very little detailed analysis. With this in mind, the important features of this presentation are the theorem and the corollaries of Section 4. The previous sections serve then as a springboard first to developing the concepts necessary to this theorem, and secondly, to acquiring a feeling for their usage.

The general scheme for this investigation is first to present a general picture of fault testing in gates. This inquiry leads to a division of all line faults on a gate into two classes, by their effects on the output function of the gate. The first class includes all faults that cause the output function to become a constant, either 1 or 0. The second class contains those faults which cause the output function to change form, but <u>not</u> become a constant.

The two forementioned classes can equally be defined by their effects on the prime implicant or implicate of the gate in question. The first class has the effect of negating the prime implicant or implicate by forcing it to become 0 or 1, respectively. Hence, a test for a fault in this class is an "existence" test, that is, it is a check on the existence of the prime implicant (implicate). The second class of faults, as discussed in this thesis, has the effect of increasing the number of minterms or maxterms of the output function (that is, as compared to the fault-free output function of the gate). Thus a test for a fault in this second class is a "growth" test, that is, a check whether the prime implicant (implicate) has changed to include more minterms (maxterms).

The above existence and growth tests are more than a compact way of defining fault tests: they are, rather, a reflection on the operation of a gate under fault conditions. They are, in a sense, a picture of how the output can change, and are therefore more important as properties of a gate. The procedure developed for selecting the necessary fault tests is, in light of what has been said, little more than a technique for checking for all the fault properties of the gate.

We believe that this is a new approach to fault testing, in that it changes the emphasis from investigating a gate under fault conditions to studying the fault properties of the gate. This change is necessary if further work is to be done concerning the development of rules for logic design so that all diagnostic tests are readily accessible or, indeed, "built in" the system.

As mentioned in the introduction, Section 3 presents an investigation of three important concepts necessary to the final theorem concerning tests for large systems. The extension of growth and existence tests to 2-level realizations, the necessity that the output expression reflect the structure of the system, and the ability to select tests which are valid for any of several irredundant forms of the same function are all germane to the theorem.

Although the theorem and corollaries are restricted to irredundant systems with only certain structures, these results are far from being impractical. The continuing use of NAND and NOR gates as building blocks for entire systems make the outcome of this paper very much in step with the state of the art.

The value of the culminating theorem as a tool for diagnosis does not affect the importance of the insight that this paper gives to the nature of faults in a system of gates.

As mentioned before, the emphasis in fault testing of gates must shift from considering the effects of faults on the operation of a gate as errors to a more practical approach of considering these effects as intrinsic properties of the gate itself.

A salient feature of this scheme for fault testing is the relative ease of test selection as compared to the more elaborate methods of path sensitizing or the unnecessarily long method of complete exhaustion of all input combinations. All the information needed for existence and growth testing is readily accessible from the circuit to be considered.

There are, however, several disadvantages or shortcomings to this technique. The first is the type of logic circuitry that can be tested. We cannot deal with such situations as AND-NAND, OR-NOR, AND-AND, and OR-OR, that is, situations where the operation of a gate is repeated by the next gate, or with networks having inverters. The second is the elaborateness of the procedures that must be used for determining a set of tests that will be valid for any form of a function with several irredundant disjunctive forms.

Several possibilities remain to be examined. Chief among these is the notion of a set of fault tests built into a logic network. Conceptually, this involves a set of rules for constructing a logic circuit such that its fault tests are available without further investigation. The ideas presented in this thesis concerning the nature of the errors due to faults are a step toward the realization of this goal.

Also, preliminary studies indicate that the technique of existence and growth testing can be extended with some modification to more complex logic circuits than those dealt with in this presentation, more specifically, those circuits with mixtures of gate types.

Finally, the extension of this technique to sequential asynchronous systems may be feasible, but this requires more elaborate discussion than is the scope of this paper.

BIBLIOGRAPHY

Armstrong, D. B., "On Finding a Nearly Minimal Set of Fault Detection Tests for Combinational Logic Nets," IEEE Transactions on Electric Computers, Vol. EC-15, No. 1, pp. 66-73 (February, 1966).

Poage, J. F., "Derivation of Optimum Tests to Detect Faults in Combinational Circuits," Proceedings of the Symposium on Mathematical Theory of Automata, pp. 483-528 (Polytechnic Press, New York, N. Y., April, 1963).

Roth, P. J., "Diagnosis of Automata Failures: A Calculus and a Method," IBM Journal, Research Division, Vol. 10, No. 4, pp. 278-291 (May, 1967).

Sellers, F. F., M. Y. Hsiao, and L. W. Bearnson, "Analyzing Errors with the Boolean Difference," IEEE Transactions on Computers, Vol. C-17, pp. 676-683 (July, 1968).

Seshu, S, "On an Improved Diagnosis Program," IEEE Transactions on Electronic Computers, Vol. EC-12, pp. 76-79 (February, 1965).

Distribution List as of April 1, 1969

Dr A.A. Dougal
Asst Director (Research)
Ofc of Defense Res & Eng
Department of Defense
Washington, D.C. 20301

Office of Deputy Director
(Research and Information, Rm 3D1037)
Department of Defense
The Pentagon
Washington, D.C. 20301

Director,Advanced Research Projects
Agency
Department of Defense
Washington, D.C. 20301

Director for Materials Sciences
Advanced Research Projects Agency
Department of Defense
Washington, D.C. 20301

Headquarters
Defense Communications Agency (340)
Washington, D.C. 20305

Defense Documentation Center
Attn: DDC-TCA
Cameron Station
Alexandria, Virginia 22314 (50 Copies)

Director
National Security Agency
Attn: TDL
Fort George G. Meade, Maryland 20755

Weapons Systems Evaluation Group
Attn: Colonel Blaine O. Vogt
400 Army-Navy Drive
Arlington, Virginia 22202

Central Intelligence Agency
Attn: OCR/DD Publications
Washington, D.C. 20505

Hq USAF (AFRDD)
The Pentagon
Washington, D.C. 20330

Hq USAF (AFRDDG)
The Pentagon
Washington, D.C. 20330

Hq USAF (AFRDSD)
The Pentagon
Washington, D.C. 20330

Colonel E.P. Gaines, Jr.
ACDA/FO
1901 Pennsylvania Ave N.W.
Washington, D.C. 20451

Lt Col R.B. Kalisch (SREE)
Chief, Electronics Division
Directorate of Engineering Sciences
Air Force Office of Scientific Research
Arlington, Virginia 22209

Dr I.R. Mirman
AFSC (SCT)
Andrews Air Force Base, Maryland 20331

AFSC (SCTSE)
Andrews Air Force Base, Maryland 20331

Mr Morton M. Pavane, Chief
AFSC Scientific and Technical Liaison Office
26 Federal Plaza, Suite 1313
New York, New York 10007

Rome Air Development Center
Attn: Documents Library (EMTLD)
Griffiss Air Force Base, New York 13440

Mr H.E. Webb (EMMIIS)
Rome Air Development Center
Griffiss Air Force Base, New York 13440

Dr L.M. Hollingsworth
AFCRL (CRN)
L.G. Hanscom Field
Bedford, Massachusetts 01730

AFCRL (CRMPLR), Stop 29
AFCRL Research Library
L.G. Hanscom Field
Bedford, Massachusetts 01730

Hq ESD (ESTI)
L.G. Hanscom Field
Bedford, Massachusetts 01730 (2 copies)

Professor J. J. D'Azzo
Dept of Electrical Engineering
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433

Dr H.V. Noble (CAVT)
Air Force Avionics Laboratory
Wright-Patterson AFB, Ohio 45433

Director
Air Force Avionics Laboratory
Wright-Patterson AFB, Ohio 45433

AFAL (AVTA/R.D. Larson
Wright-Patterson AFB, Ohio 45433

Director of Faculty Research
Department of the Air Force
U.S. Air Force Academy
Colorado Springs, Colorado 80840

Academy Library (DFSLB)
USAF Academy
Colorado Springs, Colorado 80840

Director
Aerospace Mechanics Division
Frank J. Seiler Research Laboratory (OAR)
USAF Academy
Colorado Springs Colorado 80840

Director, USAF PROJECT RAND
Via: Air Force Liaison Office
The RAND Corporation
Attn: Library D
1700 Main Street
Santa Monica, California 90045

Hq SAMSO (SMTTA/Lt Nelson)
AF Unit Post Office
Los Angeles, California 90045

Det 6, Hq OAR
Air Force Unit Post Office
Los Angeles, California 90045

AUL3T-9663
Maxwell AFB, Alabama 36112

AFETR Technical Library
(ETV,MU-135)
Patrick AFB,Florida 32925

ADTC (ADBPS-12)
Eglin AFB, Florida 32542

Mr B.R. Locke
Technical Adviser, Requirements
USAF Security Service
Kelly Air Force Base, Texas 78241

Hq AMD (AMR)
Brooks AFB, Texas 78235

USAFSAM (SMKOR)
Brooks AFB, Texas 78235

Commanding General
Attn: STEWS-RE-L, Technical Library
White Sands Missile Range
New Mexico 88002 (2 copies)

Hq AEDC (AETS)
Attn: Library/Documents
Arnold AFS, Tennessee 37389

European Office of Aerospace Research
APO New York 09667

Phsical & Engineering Sciences Division
U.S. Army Research Office
3045 Columbia Pike
Arlington, Virginia 22204

Commanding General
U.S. Army Security Agency
Attn: IARD-T
Arlington Hall Station
Arlington, Virginia 22212

Commanding General
U.S. Army Materiel Command
Attn: AMCRD-TP
Washington, D.C. 20315

Technical Director (SMUFA-A2000-107-1)
Frankford Arsenal
Philadelphia, Pennsylvania 19137

Redstone Scientific Information Center
Attn: Chief, Document Section
U.S. Army Missile Command
Redstone Arsenal, Alabama 35809

Commanding General
U.S. Army Missile Command
Attn: AMSMI-REX
Redstone Arsenal, Alabama 35809

Commanding General
U.S. Army Strategic Communications Command
Attn: SCC-CG-SAE
Fort Huachuca, Arizona 85613

Commanding Officer
Army Materials and Mechanics Res. Center
Attn: Dr H. Priest
Watertown Arsenal
Watertown, Massachusetts 02172

Commandant
U.S. Army Air Defense School
Attn: Missile Science Division, C&S Dept
P.O. Box 9390
Fort Bliss, Texas 79916

Commandant
U.S. Army Command & General Staff College
Attn: Acquisitions, Library Division
Fort Leavenworth, Kansas 66027

Commanding Officer
U.S. Army Electronics R&D Activity
White Sands Missile Range, New Mexico 88002

Mr Norman J. Field, AMSEL-RD-S
Chief, Office of Science & Technology
Research and Development Directorate
U.S. Army Electronics Command
Fort Monmouth, New Jersey 07703

Commanding Officer
Harry Diamond Laboratories
Attn: Dr Berthold Altman (AMXDO-TI)
Connecticut Avenue and Van Ness St N.W.
Washington, D.C. 20438

Director
Walter Reed Army Institute of Research
Walter Reed Army Medical Center
Washington, D.C. 20012

Commanding Officer (AMXRD-BAT)
U.S. Army Ballistics Research Laboratory
Aberdeen Proving Ground
Aberdeen, Maryland 21005

Technical Director
U.S. Army Limited War Laboratory
Aberdeen Proving Ground
Aberdeen, Maryland 21005

Commanding Officer
Human Engineering Laboratories
Aberdeen Proving Ground
Aberdeen, Maryland 21005

U.S. Army Munitions Command
Attn: Science & Technology Br. Bldg 59
Picatinny Arsenal, SMUPA-VA6
Dover, New Jersey 07801

U.S. Army Mobility Equipment Research
and Development Center
Attn: Technical Document Center, Bldg 315
Fort Belvoir, Virginia 22060

Director
U.S. Army Engineer Geodesy,
Intelligence & Mapping
Research and Development Agency
Fort Belvoir, Virginia 22060

Dr Herman Robl
Deputy Chief Scientist
U.S. Army Research Office (Durham)
Box CM, Duke Station
Durham, North Carolina 27706

Richard O. Ulsh (CRDARD-IPO)
U.S. Army Research Office (Durham)
Box CM, Duke Station
Durham, North Carolina 27706

Mr Robert O. Parker, AMSEL-RD-S
Executive Secretary, JSTAC
U.S. Army Electronics Command
Fort Monmouth, New Jersey 07703

Commanding General
U.S. Army Electronics Command
Fort Monmouth, New Jersey 07703
Attention:AMSEL-SC
RD-GF
RD-MT
XL-D
XL-E
XL-C
XL-S (Dr R. Buser)
HL-CT-DD
HL-CT-R
HL-CT-L (Dr W.S. McAfee)
HL-CT-O
HL-CT-I
HL-CT-A
NL-D
NL-A
NL-P
NL-P-2 (Mr D. Haratz)
NL-R (Mr R. Kulinyi)
NL-S
KL-D
KL-E
KL-S (Dr H. Jacobs)
KL-SM (Drs Schiel/Hieslmair)
KL-T
VL-D
VL-F (Mr R.J. Niemela)
WL-D

1 copy to each symbol listed individually addressed

Dr A.D. Schnitzler, AMSEL-HL-NVII
Night Vision Laboratory, USAECOM
Fort Belvoir, Virginia 22060

Dr G.M. Janney, AMSEL-HL-NVOR
Night Vison Laboratory,USAECOM
Fort Belvoir, Virginia 22060

Atmospheric Sciences Office
Atmospheric Sciences Laboratory
White Sands Missile Range
New Mexico 88002

Missile Electronic Warfare,
Technical Area, AMSEL-WT-MT
White Sands Missile Range
New Mexico 88002

Project Manager
Common Positioning & Navigation Systems
Attn: Harold H. Bahr (AMCPM-NS-TM), Bldg 439
U.S. Army Electronics Command
Fort Monmouth, New Jersey 07703

Director, Electronic Programs
Attn: Code 427
Department of the Navy
Washington, D.C. 20360

Commander
U.S. Naval Security Group Command
Attn: G43
3801 Nebraska Avenue
Washington, D.C. 20390

Director
Naval Research Laboratory
Washington, D.C. 20390
Attn: Code 2027 6 copies
Dr W.C. Hall, Code 7000 1 copy
Dr A. Brodizinsky, Sup.Elec Div. 1 copy

Dr G.M.R. Winkler
Director, Time Service Division
U.S. Naval Observatory
Washington, D.C. 20390

Naval Air Systems Command
AIR 03
Washington, D.C. 20360 2 copies

Naval Ship Systems Command
Ship 031
Washington, D.C. 20360

Naval ship Systems Command
Ship 035
Washington, D.C. 20360

U.S. Naval Weapons Laboratory
Dahlgren, Virginia 22448

Naval Electronic Systems Command
ELEX 03, Room 2046 Munitions Building
Department of the Navy
Washington, D.C. 20360 (2 copies)

Commander
Naval Electronics Laboratory Center
Attn: Library
San Diego, California 92152 (2 copies)

Deputy Director and Chief Scientist
Office of Naval Research Branch Office
1030 Est Gree Street
Pasadena, California 91101

Library (Code2124)
Technical Report Section
Naval Postgraduate School
Monterey, California 93940

Glen A. Myers (Code 52Mv)
Assoc Professor of Elec. Engineering
Naval Postgraduate School
Monterey, California 93940

Commanding Officer and Director
U.S. Naval Underwater Sound Laboratory
Fort Trumbull
New London, Connecticut 06840

Commanding Officer
Naval Avionics Facility
Indianapolis, Indiana 46241

Dr H. Harrison, Code RRE
Chief, Electrophysics Branch
National Aeronautics & Space Admin.
Washington, D.C. 20546

NASA Lewis Research Center
Attn: Library
21000 Brookpark Road
Cleveland, Ohio 44135

Los Alamos Scientific Laboratory
Attn: Report Library
P.O. Box 1663
Los Alamos, New Mexico 87544

Federal Aviation Administration
Attn: Admin Stds Div (MS-110)
800 Independence Ave S.W.
Washington, D.C. 20590

Head, Technical Services Division
Naval Investigative Service Headquarters
4420 North Fairfax Drive
Arlington, Virginia 22203

Commander
U.S. Naval Ordnance Laboratory
Attn: Librarian
White Oak, Maryland 21502 (2 copies)

Commanding Officer
Office of Naval Research Branch Office
Box 39 FPO
New York, New York 09510

Commanding Officer
Office of Naval Research Branch Office
219 South Dearborn Street
Chicago, Illinois 60604

Commanding Officer
Office of Naval Research Branch Office
495 Summer Street
Boston, Massachusetts 02210

Commander (ADL)
Naval Air Development Center
Johnsville, Warminster, Pa 18974

Commanding Officer
Naval Training Device Center
Orlando, Florida 32813

Commander (Code 753)
Naval Weapons Center
Attn: Technical Library
China Lake, California 93555

Commanding Officer
Naval Weapons Center
Corona Laboratories
Attn: Library
Corona, California 91720

Commander, U.S. Naval Missile Center
Point Mugu, California 93041

W.A. Eberspacher, Associate Head
Systems Integration Division
Code 5340A, Box 15
U.S. Naval Missile Center
Point Mugu, California 93041

Mr M. Zane Thornton, Chief
Network Engineering, Communications
and Operations Branch
Lister Hill National Center for
Biomedical Communications
8600 Rockville Pike
Bethesda, Maryland 20014

U.S. Post Office Department
Library - Room 1012
12th & Pennsylvania Ave, N.W.
Washington, D.C. 20260

Director
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Mr Jerome Fox, Research Coordinator
Polytechnic Institute of Brooklyn
55 Johnson Street
Brooklyn, New York 11201

Director
Columbia Radiation Laboratory
Columbia University
538 West 120th Street
New York, New York 10027

Director
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

Director
Stanford Electronics Laboratories
Stanford University
Stanford, California 94305

Director
Microwave Physics Laboratory
Stanford University
Stanford, California 94305

Director, Electronics Research Laboratory
University of California
Berkeley, California 94720

Director
Electronic Sciences Laboratory
University of Southern California
Los Angeles, California 90007

Director
Electronics Research Center
The University of Texas at Austin
Austin Texas 78712

Division of Engineering and Applied Physics
210 Pierce Hall
Harvard University
Cambridge, Massachusetts 02138

Dr G.J. Murphy
The Technological Institute
Northwestern University
Evanston, Illinois 60201

Dr John C. Hancock, Head
School of Electrical Engineering
Purdue University
Lafayette, Indiana 47907

Dept of Electrical Engineering
Texas Technological College
Lubbock, Texas 79409

Aerospace Corporation
P.O. Box 95085
Los Angeles, California 90045
Attn: Library Acquisitions Group

Proffessor Nicholas George
California Inst of Technology
Pasadena, California 91109

Aeronautics Library
Graduat Aeronautical Laboratories
California Institute of Technology
1201 E. California Blvd
Pasadena, California 91109

The John Hoplins University
Applied Physics Laboratory
Attn: Document Librarian
8621 Georgia Avenue
Silver Spring, Maryland 20910

Raytheon Company
Attn: Librarian
Bedford, Massachusetts 01730

Raytheon Company
Research Division Library
28 Seyon Street
Waltham, Massachusetts 02154

Dr Sheldon J. Wells
Electronic Properties Information Center
Mail Station E-175
Hughes Aircraft Company
Culver City, California 90230

Dr Robert E. Fontana
Systems Research Laboratories Inc.
7001 Indian Ripple Road
Dayton, Ohio 45440

Nuclear Instrumentation Group
Bldg 29, Room 101
Lawrence Radiation Laboratory
University of California
Berkeley, California 94720

Sylvania Electronic Systems
Applied Research Laboratory
Attn: Documents Librarian
40 Sylvan Road
Waltham, Massachusetts 02154

Hollander Associates
P.O. Box 2276
Fullerton, California 92633

Illinois Institute of Technology
Dept of Electrical Engineering
Chicago, Illinois 60616

The University of Arizona
Dept of Electrical Engineering
Tucson, Arizona 85721

Utah State University
Dept Of Electrical Engineering
Logan, Utah 84321

Case Institute of Technology
Engineering Division
University Circle
Cleveland, Ohio 44106

Hunt Library
Carnegie-Mellon University
Schenley Park
Pittsburgh, Pennsylvania 15213

Dr Leo Youns
Stanford Research Institute
Menlo Park, California 94025

School of Engineering Sciences
Arizona State University
Tempe, Arizona 85281

Engineering & Mathmatical Sciences Library
University of California at Los Angeles
405 Hilgred Avenue
Los Angeles, California 90024

The Library
Government Publications Section
University of California
Santa Barbara, California 93106

Carnegie Institute of Technology
Electrical Engineering Department
Pittsburgh, Pennsylvania 15213

Professor Joseph E. Rowe
Chairman, Dept of Electrical Engineering
The University of Michigan
Ann Arbor, Michigan 48104

New York University
College of Engineering
New York, New York 10019

Syracuse University
Dept of Electrical Engineering
Syracuse, New York 13210

Yale University
Engineering Department
New Haven, Connecticut 06520

Airborne Instruments Laboratory
Deerpark, New York 11729

Raytheon Company
Attn: Librarian
Bedford, Massachusetts 01730

Lincoln Laboratory
Massachusetts Institute of Technology
Lexington, Massachusetts 02173

The University of Iowa
The University Libraries
Iowa City, Iowa 52240

Lenkurt Electric Co,Inc
1105 County Road
San Carlos, California 94070
Attn: Mr E.K. Peterson

Philco Ford Corporation
Communications & Electronics Div.
Union Meeting and Jolly Rods
Blue Bell, Pennsylvania 19422

Union Carbide Corporation
Electronic Division
P.O. Box 1209
Mountain View, California 94041

Electromagnetic Compatibility Analysis Center
(ECAC), Attn: ACLP
North Severn
Annapolis, Maryland 21402

Director
U. S. Army Advanced Materiel Concepts Agency
Washington, D.C. 20315

ADDENDUM

Dept of Electrical Engineering
Rice University
Houston, Texas 77001

Research Laboratories for the Eng. Sciences
School of Engineering & Applied Science
University of Virginia
Charlottesville, Virginia 22903

Dept of Electrical Engineering
College of Engineering & Technology
Ohio University
Athens, Ohio 45701

Project MAC
Document Room
Massachusetts Institute of Technology
545 Technology Square
Cambridge, Massachusetts 02139

ERRATUM

Mr Jerome Fox, Research Coordinator
Polytechnic Institute of Brooklyn
55 Johnson Street (should be 333 Jay Street)
Brooklyn, N.Y. 11201

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*

University of Illinois
Coordinated Science Laboratory
Urbana, Illinois 61801

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

GENERATION OF DIAGNOSTIC TESTS USING PRIME IMPLICANTS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

PAIGE, Michael Reid

6. REPORT DATE

May, 1969

7a. TOTAL NO. OF PAGES

36

7b. NO. OF REFS

5

8a. CONTRACT OR GRANT NO.

DAAB 07-67-C-0199; also NSF GK-1663

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

R-414

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

This Document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Joint Services Electronics Program
thru U.S. Army Electronics Command
Fort Monmouth, New Jersey 07703

13. ABSTRACT

NONE

DD FORM 1 NOV 65 **1473**

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Fault Detection						
Prime Implicant						
Existence Test						
Growth Test						