Functions for improving diagnostic resolution in an LSI environment*

by MADHUKUMAR A. MEHTA

GTE Automatic Electric Laboratories Incorporated
Northlake, Illinois

and

HENRY P. MESSINGER

Illinois Institute of Technology
Chicago, Illinois

and

WILLIAM B. SMITH

Bell Telephone Laboratories Incorporated
Holmdel, New Jersey

INTRODUCTION

LSI implementation of digital circuitry opens the door to the consideration of dramatically new approaches to the design of system fault diagnosis. New constraints have been added, such as the difficulty of inserting test access points internal to large pieces of circuitry. At the same time, failure modes seem to be changing with bonding lead failures increasing in importance. This paper presents an approach that leans heavily on the assumption that adding additional logic to a circuit is of little consequence, whereas it is important to reduce the access provided for testing capability. As the practicality of the proposed approach has not been examined in detail, the concept is primarily presented to stimulate further study into the special problems and opportunities involved in diagnosis of LSI systems.

The approach discussed here is to incorporate on each least replaceable unit (LRU) a special combinational test circuit that will identify, by observation of the output alone, any stuck at "0" or "1" input failures to the particular input involved. Such a test circuit is termed an "Ambiguity Resolver" (AR) function, and proof of its existence for any number of inputs will be presented. The identification of a failure to a particular input in general actually only isolates the failure to the bus that is connected to that input because of the propagating tendency of failures on a bus. Thus, all LRU's connected to the bus implicated are equal candidates to have failures on their bus connections. A capability such as provided by AR functions would obviously be of little use in a situation in which buses tend to be common to a group of LRU's. However, when the observable points are restricted to the functional outputs, conventional methods usually do not provide resolution of input/output faults to a unique bus. This feature is conveniently provided by the usage of AR functions.

Let us define some of the terms and clarify the notations used in this article.

DEFINITIONS AND NOTATIONS

Observable Points: points at which the outcome of an applied test procedure can be observed.
Failure Exclusive Points: a set of points only one of which is affected by any one assumed failure.

Independent Inputs: two inputs are independent if (and only if) the logic state of either can be changed without affecting the logic state of the other.

Sensitized Path: a path between a point in a combinational circuit and the output is sensitized if a change of logic conditions at that point results in a change of logic conditions at the output while other inputs are not changed.

Output Vector: the ordered set of (binary) outputs that result from the application of a test procedure (a sequence of tests). The output vector corresponding to no-fault conditions is called the normal output vector and is denoted by $\Sigma_n$; also $\Sigma_f$ denotes the output vector for the same test procedure when fault $f$ is present.

Complementary Failures: Stuck-at-0 (s-a-0) and Stuck-at-1 (s-a-1) failures at a particular point (input or output) are called complementary.

Least Replaceable Unit (LRU): the smallest subsystem which will be completely replaced if a fault is located at the terminals of, or inside, the subsystem.

Capital letters of the English alphabet are used to denote the inputs and the outputs of a circuit. Usually the early letters ($A, B, C$) are used for the inputs while the later letters are reserved for the outputs ($Z, Y, X, W$). Exceptions in these notations are made when dealing with networks to avoid duplicate labeling for connections (buses). Subscripts are used to identify the inputs or the outputs. A distinction is made between the inputs to the circuit ($A_i$) and the leads which carry these inputs to the circuit. Small letters with identical subscripts ($a_i$) designate the corresponding leads for any input. Similar notations are used to distinguish the outputs and the output leads. A fault at the input lead is fed back to the input (output of the driving gate) only if it is a propagating fault (i.e., if it affects the entire bus). A lead $a_i$ with a s-a-0 fault, in tables and figures, is denoted by symbol $a_i^0$; $a_i^1$ similarly denotes the same lead with a s-a-1 fault.

AMBIGUITY RESOLVER (AR) FUNCTION

Since LSI implementation makes redundant logic economic to use, we examine the possible ways of using redundancy (internal to a LSI circuit) to simplify fault diagnosis of digital systems.

One of the ways to use the redundancy is the conventional approach of duplicate and compare (Figure 1). Two disadvantages prevent this scheme from being attractive. First, the required redundancy is excessive; it takes more than double the circuitry to implement this approach. Second, the approach simplifies the diagnosis of only the internal faults. For the diagnosis of the faults at the bonding leads one has to still resort to other approaches. Unfortunately, a predominant failure mode in systems composed of LSI, that have been in systems which are in operation for awhile, is due to bond failures. Therefore, the approach of
Figure 1 does not represent an efficient usage of redundancy.

The proposed approach (Figure 2) consists of a specialized single output test function \((F)\) for simplifying the diagnosis of input faults. These Ambiguity Resolver (AR) functions are combinational logic functions which have the property that input/output faults can be resolved to a unique bus through the observation of the output alone. We will show that such functions exist and that they are simple in comparison to the normal function \((F_I)\). AR functions simplify the fault diagnosis of digital systems implemented with LSI circuits because:

1. A predominant failure mode in operating IC’s is bond failures which result in input/output faults. The AR functions simplify the diagnosis of only the input faults; however, the output faults will become input faults of the driven LRU’s and therefore will be diagnosed by their AR functions. It is reasonable to assume that the faults of the unused outputs do not affect the system performance.

2. AR functions permit resolution of input/output faults to a unique bus. Such resolution is not always available by the use of conventional test procedures when the observable points are restricted to the functional outputs.

3. Only one point need be observed for diagnosis.

4. Since the AR function \((F)\) is unrelated to the basic function \(F_I\), its design is completely independent; for example, \(F_I\) may be a sequential function, yet \(F\) can be a simple combinational function. Due to this independence, often diagnosis using AR functions requires fewer tests in comparison to a conventional test procedure.

Although the AR approach simplifies fault diagnosis of LSI implemented systems, it is certainly not the complete answer. For example, the AR approach requires testing, and thus does not provide an on-line detection of faults. Also AR function provides neither fault detection nor diagnosis of faults within an LRU. Therefore, in any practical application, the AR approach would probably be augmented with other, more conventional methods.

Since the AR function can be independently designed, in the following sections, we will disregard the basic function. We will concern ourselves only with single output combinational functions which have the AR property. We will assume:

1. Only input/output faults.
2. Only s-a-I, s-a-O faults.
3. One fault at a time.
4. Access to only the inputs of the function. That is, only the inputs can be controlled externally.
5. Output is the only observable point, i.e., the performance of the function, during the test procedure, can be observed only at the output.

In order to clarify some of the concepts presented here, Section 8 presents an example of the use of AR functions along with a conventional scheme for comparison.

COMPLEMENTARY FAILURES

Understanding the constraints that are imposed on circuits being tested is required before we can go into a discussion of the existence of AR functions. An important constraint is on the allowable outputs for complementary input failures.

Consider a test \(t_i\) that applies the input \(a_{i1}a_{i2} \ldots a_{is}\) to a circuit, and consider a particular input lead \(a_k\) (see Table I). If the output of the circuit is \(\sigma_i\) under the test \(t_i\), then the output can be either \(\sigma_i\) or \(\bar{\sigma}_i\) for \(a_k\) s-a-0 and similarly for \(a_k\) s-a-1. If the output is \(\sigma_i\) for \(a_k\) both s-a-0 and s-a-1, then the output is fault insensitive to \(a_k\). If the output is \(\bar{\sigma}_i\) for \(a_k\) s-a-0 and \(\sigma_i\) for \(a_k\) s-a-1, then it is clear that \(a_k\) must have a “1” input for this test \(t_i\). A similar story is true for output \(\sigma_i\) for \(a_k\) s-a-0 and \(\bar{\sigma}_i\) for \(a_k\) s-a-1 (\(a_k\) has a “0” input). However, it is not possible to have an output \(\bar{\sigma}_i\) for both kinds of failures because this would imply that \(\bar{\sigma}_i\) is always the

<p>| Table I—Outcomes of a Test for Complementary Failures at the Input (a_k) and their Implications |</p>
<table>
<thead>
<tr>
<th>Test Input Conditions</th>
<th>Normal Possibility</th>
<th>Implications for the test</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A_{i1-i_{n}}) Output</td>
<td>(\sigma_i)</td>
<td>(a_k) = 0 or 1 fault insensitive</td>
</tr>
<tr>
<td>(a_{i1}a_{i2} \ldots a_{is})</td>
<td>(\sigma_i)</td>
<td>(a_k) = 0</td>
</tr>
<tr>
<td>(a_{i1}a_{i2} \ldots a_{is})</td>
<td>(\bar{\sigma}_i)</td>
<td>(a_k) = 1</td>
</tr>
</tbody>
</table>

From the collection of the Computer History Museum (www.computerhistory.org)
output for \( a_k \) with either 0 or 1 input, which contradicts the original assumption. Theorems 1 and 2 state the results just given, which are proved elsewhere in detail. They are given below because the results will be used later.

**Theorem 1:**

The outcome of a test \( t_i \) cannot be different from normal case under both a s-a-0 and a s-a-1 failure at the same input.

**Theorem 2:**

If the outcome of the test \( t_i \) is different from the normal case for the fault \( f_i \), then the outcome of the test \( t_i \) for the complementary failure is identical to the normal case.

Finally, each detectable failure must produce a non-normal output for at least one of the tests. For that test, by Theorem 2, the complementary failure must have normal output. Therefore, the resulting output vectors for complementary failures cannot be the same. Thus, although this is not necessary for localization of input faults to a unique bus, complementary failures of AR function will always be distinguishable. Having established this, we are now in a position to formally define the AR function.

**DEFINITION AND PROPERTIES OF AR FUNCTIONS**

**Definition:** An \( n \)-input AR function is a combinational function of \( n \) variables with a single output with the following properties:

1. All faults are detectable, i.e., there exists a test procedure \( T \) containing \( p \) tests such that \( 1 \leq p \leq 2^n \) and \( \Sigma_i \neq \Sigma_i \) for any fault \( f_i \).
2. All faults produce unique output vectors for the test procedure \( T \), i.e., \( \Sigma_i = \Sigma_i \) iff \( f_i = f_i \).

Any such set of \( p \) tests will be referred to as a Test Pattern of the AR function (under the fault assumptions made earlier).

Some elementary properties are easily derived from the basic definition of the AR function. Clearly, the output faults yield either an all 0's (s-a-0) or an all 1's (s-a-1) output vector. Also, the normal output vector for the test pattern has to be non-zero and non-all 1's since output faults are to produce non-normal output vectors. Finally, each input fault must yield a unique, non-zero and non-all 1's output vector.

**Theorem 3:**

If the permissible failure modes include complementary faults at the inputs and at the outputs, then the complement of the normal output vector for an AR function is different from the resulting output vectors for all permissible failures.

**Proof:**

Since the normal output vector is non-zero, non-all 1's, the complement of the normal output vector \( (\Sigma_n) \) is also non-zero, non-all 1's. Thus \( \Sigma_n \) is different from the resulting output vectors for the output faults.

Next assume that the resulting output vector for an input fault \( f_i \) is identical to \( \Sigma_n \). This implies that the outcome for each test \( t_j \) \((1 \leq j \leq p) \) for the fault \( f_i \) is different from the normal case. Now consider the complementary failure \( f_i^c \). By Theorem 1, for any test \( t_j \) the outcome for both failures at the same input cannot have different performance from the normal. Therefore (by Theorem 2), the outcome for each test \( t_j \) for the fault \( f_i^c \) must be identical to the normal case. Thus the resulting output vector for the fault \( f_i \) for the test pattern \( T \) will be identical to the normal case. But this is a contradiction since the fault \( f_i \) during the test pattern \( T \) must produce a unique output vector different from the normal case. Thus the resulting output vector of an input fault of an AR function also cannot be identical to \( \Sigma_n \). (Q.E.D.)

Next we examine other properties of AR functions which will permit synthesis of new AR functions from

![Figure 3a—Not conformal because faults at a1 violate condition 1](From the collection of the Computer History Museum (www.computerhistory.org))
the available AR functions. It will be shown that the resulting functions have the AR property when certain structural constraints are observed. The concept of conformal structure, introduced next, provides one convenient way to produce new AR functions. It should be noted, however, that although conformal structures, as defined below, have the AR property, structures having the AR property need not be conformal.

Definition:

When a one output combinational function is generated by the use of AR functions and logic gates, the resulting structure is conformal iff

1. Each input failure of the resulting function results in a permissible input failure at one, and only one, of the AR functions,
2. No two input failures of the resulting function result in an identical input failure condition at any AR function,
3. Application of the test pattern for each AR function is possible by control of the inputs of the resulting function, and
4. For the output of each AR function, a sensitized path to the output of the resulting function exists.

Figure 3, where $F_1$ and $F_2$ are AR functions, illustrates a number of structures which are not conformal for one or more reasons.

The motivation for the use of various conditions in the above definition can be intuitively explained at this point. Condition 1 insures that each input fault can be made distinguishable from the other faults by the test pattern of one of the AR functions. The second part of the condition ("only one") prevents one input fault

From the collection of the Computer History Museum (www.computerhistory.org)
from affecting more than one AR function. The resulting output vectors of each AR function need a sensitized path to the final output to be observable under normal, as well as under various failure conditions. One of the ways to provide such a path is to maintain a unique combination of the inputs at the remaining AR functions (which provides a sensitized path) during the testing of each AR function. Following this procedure, in addition, permits distinguishability of the input faults of one AR function from the input faults of other AR functions as we will see. If an input fault of the resulting structure affected more than one AR function, this procedure may not be possible in all cases. Condition 2 prevents multiple input failures from having identical output vectors, thereby permitting the required localization. Condition 3 assures the ability to apply test patterns to various AR functions, and the condition 4 makes the resulting output vector available at the final output of the resulting structure.

Having established some structural constraints and some intuitive basis for them, we proceed to prove the AR property of conformal structures.

**Theorem 4:**

A conformal structure implemented with AR functions and logic gates results in an AR function.

**Proof:**

Let $F_1, F_2, \ldots, F_n$ be the $n$ AR functions used in the conformal structure. We will refer to these functions as component AR functions. Let $Z_i$ be the output of the function $F_i$, and let $Z$ be the output of the resulting function $F$ with conformal structure.

To prove the AR property of the resulting conformal structure, we will show the existence of a test procedure which provides the distinguishability of the input/output faults of $F$.

As the structure is conformal, each input fault of the resulting structure results in a permissible input fault at one (and only one) AR function. Thus the set of input faults of the resulting structure is a subset of the input faults of the component AR functions. Also due to the second condition of conformality, these input faults of $F$ produce unique input faults at the component AR functions. Thus if the input faults of the component AR functions are distinguishable, then the input faults of the resulting function are also distinguishable.

Consider a test procedure wherein we apply the test pattern to each component AR function, one after another, by control of some of the inputs while maintaining the remaining inputs at some logic conditions so as to provide a sensitized path to the output $Z$ throughout the test pattern. The conformality of the structure accounts for the feasibility of such a test procedure. We will show that this test procedure provides the required distinguishability.

To show the distinguishability, we will follow these steps. First we will show that input faults of any $F_i$ are distinguishable from the other input faults of the same AR function. Next we will show that these faults are distinguishable from the faults at the output $Z$. Then we will show the distinguishability of these faults from the input faults of the other AR functions. Note that the distinguishability is symmetric, i.e., if $a$ is distinguishable from $b$, then $b$ is distinguishable from $a$. Therefore, the distinguishability of the output $(Z)$ faults from the input faults of the resulting structure is easily established. Finally, we will show the distinguishability of the output faults from each other.

During the test procedure, the test pattern $T_i$ is applied to $F_i$, and a sensitized path to the final output for the output $Z_i$ is provided. As $F_i$ is an AR function, its input faults produce unique non-normal output vectors at $Z_i$ for the test pattern $T_i$, which due to the availability of the sensitized path are transmitted to the output $Z$. (The normal output at $Z$ may be the same as, or could be the complement of, the output at $Z_i$ during the test pattern $T_i$, as sensitization does not guarantee the polarity of the output. But as the uniqueness is what we are concerned with, and as uniqueness is preserved under complementation, we do not worry about the polarity.) Note that due to the conformality of the structure during the test pattern $T_i$, only permissible faults (s-a-0 or s-a-1) exist at the inputs of the function $F_i$. Further, as we consider only a single failure, when we are considering the distinguishability of the two input faults of $F_i$, the path sensitization can be assumed to be failure free. Thus the input faults of $F_i$ produce unique output vectors at the output $Z_i$, which, in turn, produce unique output vectors at the output $Z$. Therefore, the input faults of $F_i$ are distinguishable from the other input faults of the same AR function $F_0$, when test pattern $T_i$ is applied as a part of the proposed test procedure by observing $Z$. Also, as the resulting output vector at the output $Z$ for each input fault is different from normal, the input faults are detectable.

Now consider the distinguishability of the input faults of the function $F_i$, from the input faults at the output $Z$. Observe that when the test pattern $T_i$ is applied, as a part of the test procedure, the observed output vector for the faults at the output $Z$ will be either all
functions are detectable and distinguishable from one another.

Therefore, as the test procedure considered, provides unique, non-normal, output vectors for various faults, the resulting conformal structure has the AR property and the test procedure is its test pattern. (Q.E.D.)

It becomes apparent that the formulation of conformal structure provides us with a great flexibility in generating new AR functions. Of these, three are worthy of specific mention.

**Corollary 1:** The complement of an AR function is an AR function.

**Corollary 2:** The AND function of AR functions is an AR function if the resulting structure is conformal.

**Corollary 3:** The OR function of AR functions is an AR function if the resulting structure is conformal.

**EXISTENCE OF A GENERAL N-INPUT AR FUNCTION**

There are a number of ways by which we can prove the existence of an n-input AR function, for any n. For instance, it is easy to show that a parity network* with any number of inputs has the AR property. However, for three or more inputs other functions, apparently less complex, also exhibit the AR property. We, therefore, will create simple two- and three-input AR functions and use them in a constructional proof for proving the existence of a general AR function. This proof is preferred because it also provides a scheme for generating relatively simple AR functions.

**Existence of a 2-input AR function**

Figure 4 shows a two input exclusive OR function. Also shown are the resulting functions for various faults. Note the uniqueness of the resulting function for each fault. As the resulting function for each fault is unique and non-normal the exclusive OR function is an AR function. Table II shows a test pattern for the AR function. That this is a test pattern can be shown by generating the resulting output vectors for various conditions and showing their uniqueness.

**Existence of a 3-input AR function**

Figure 5 shows a three input function. Also shown are the resulting functions for various faults. As the result-
**Figure 4—A 2-Input AR function**

\[ Z_1 \text{(NORMAL)} = a_1 \overline{a}_2 + \overline{a}_1 a_2 \]

\[ Z_1 (z_1^0) = 0 \quad Z_1 (a_1^1) = \overline{a}_2 \]

\[ Z_1 (z_1^1) = 1 \quad Z_1 (a_2^0) = a_1 \]

\[ Z_1 (a_1^0) = a_2 \quad Z_1 (a_2^1) = \overline{a}_1 \]

**Theorem 5:**

There exists an \( n \)-input AR function for any finite \( n \geq 2 \).

**Proof:**

We have already shown the existence of 2-input and 3-input AR functions. Consider any integer \( n > 3 \), then it is always possible to find integers \( q \) and \( r \) (\( 2 \leq q \leq n/2 \) and \( 0 \leq r \leq 1 \)) such that \( n = 2q + 3r \). In other words, for all even integers, we have \( r = 0 \), and \( q = (n/2) \) while for odd integers \( r = 1 \) and \( q = [(n-3)/2] \).

Since the OR function of AR functions is an AR function by Theorem 4 (Corollary 3), we can generate an \( n \)-input AR function for \( n > 3 \) by generating an OR function of \( q \) 2-input AR functions and \( r \) 3-input AR functions with conformal structure. Therefore, \( n \)-input AR function exists for any finite \( n \geq 2 \). (Q.E.D.)

**THE CONVERGENCE PROPERTY**

In the previous sections, we examined few techniques to generate new AR functions. The functions, thus

**TABLE II—A Test Pattern for the 2-Input AR Function of Figure 4**

<table>
<thead>
<tr>
<th>Test No.</th>
<th>Inputs A₁</th>
<th>A₂</th>
<th>Normal Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**TABLE III—A Test Pattern for the 3-Input AR Function of Figure 5**

<table>
<thead>
<tr>
<th>Test No.</th>
<th>Inputs A₁</th>
<th>A₂</th>
<th>A₃</th>
<th>Normal Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

From the collection of the Computer History Museum (www.computerhistory.org)
generated, had the property of failure localization to a single bus for the input/output faults of the resulting function. To that extent, these new functions can be used internal to the LRU to provide the failure localization capability. If we provide an AR function in each LRU, the special output (the output of the AR function) of each LRU must be made observable. Does the possibility of reducing the total number of observable points exist? To answer this, we consider if convergence of AR functions by an AR function preserves the failure localization property.

First, let us illustrate what we mean by a converging connection. Let \( F_1, F_2, \ldots, F_n \) be \( n \) AR functions. Further, let each \( F_i \) have \( m_i \) inputs and one output. Also let \( F_c \) be an AR function with \( n \) inputs. Now consider a configuration, shown in Figure 6, where the output \( Z_i \) of each \( F_i \) is connected to the \( i \)th input of \( F_c \). Let \( Z \) be the output of \( F_c \). The resulting structure will be termed a converging connection.

It can be shown that when the inputs to a converging connection are independent and failure exclusive, the input/output faults of each component AR function \( (F_i,s \text{ or } F_c) \) are locatable by observing \( Z \) alone. Thus a test procedure exists which produces unique output vectors at the final output \( Z \) for each of the faults at the inputs \( (a_{ij}; 1 \leq i \leq n, 1 \leq j \leq m_i) \) or at the outputs \( (Z_i \text{ or } Z_c) \). Further, it can be shown that the length of such a test procedure does not exceed the sum of the tests in the test patterns of the component AR functions. The detailed proof of this result, which follows a similar line of reasoning as that of Theorem 4, is rather lengthy and is, therefore, not included.

Next we demonstrate the application of converging connections in reducing the number of points which must be observable. Note that in Figure 6 if all AR functions \( F_i \) were used in different LRU, all of the \( n \) outputs \( Z_i \)'s must be made observable. In that case, the test pattern \( T_i \); however, can be applied simultaneously to each AR function \( F_i \). Thus the number of tests would be equal to the maximum of the number of tests in the test patterns \( T_i \)'s. On the other hand, by the use of a converging connection we are able to reduce the number of points which must be made observable to one (from \( n \)). But the length of the test procedure to obtain the same degree of diagnostic resolution is now equal to (or less than) the sum of the number of tests in all the test patterns \( T_i \) \((1 \leq i \leq n)\) and the number of tests in the test pattern \( T_c \) for the AR function \( F_c \).

AN ILLUSTRATIVE EXAMPLE

To illustrate the application of AR functions, a very elemental LRU (Figure 7) is selected as a building block of a simple subsystem (Figure 8). Realistic systems, of course, would be considerably more complex. A comparison is made of the AR approach with present day techniques in terms of the required number of
The points which are to be observable are identified by the arrowhead.

observable points and the length of the required test procedure.

A conventional approach

As can be easily inferred from the functions for $Z_{31}$ and $Z_{32}$, if the observable points are restricted to the final outputs ($Z_{31}$ and $Z_{32}$), all faults cannot be localized. For instance, the s-a-0 faults at $a_{11}$, $a_{12}$, and $a_{13}$ and s-a-1 fault at $z_{11}$ are indistinguishable. Similarly, the s-a-1 faults at $a_{21}$, $a_{22}$, and $a_{23}$ and s-a-0 fault at $z_{21}$ are indistinguishable. The distinguishability of these faults can be obtained if the outputs $Z_{12}$ and $Z_{22}$ are made observable. Thus to obtain a complete localization we need four observable points.

Table IV shows a conventional test procedure for the subsystem generated by the use of well-known techniques. The first four tests provide the testing of the faults at the inputs of $F_{13}$. The conditions at the other inputs are maintained as prescribed to provide a sensitized path to the output $Z_{31}$. One of these tests also provides an all zero test for $F_{12}$. The next three tests similarly provide the testing of the faults at the inputs of $F_{13}$. Note that the input combinations for $F_{13}$ to provide a sensitized path to the output $Z_{32}$ during these tests, are carefully chosen to simultaneously provide the tests for $F_{12}$ when the output $Z_{32}$ is observable. The eighth test provides partly for the testing of the input $a_{31}$. Observe that so far only a subset of the total tests required is applied (indirectly) to $F_{13}$. The conditions corresponding to the remaining tests are applied by the tests 9 through 12. Note the use of carefully chosen input combination at the inputs of $F_{12}$ to simultaneously provide the required test combinations if the output $Z_{32}$ was observable. Finally test 13 is necessary to provide the remaining test combination for $F_{13}$ when output $Z_{31}$ is made observable. Thus we see that a total of thirteen tests and four observable points are required for localization of all faults.

The proposed approach

Next, consider the use of a three-input AR function ($F_{AR}$) as shown in Figure 9, which is the function discussed in Section 6. One output is to be added to the LRU to make the output of the AR function observable.

Consider the implementation of the subsystem using LRU's with AR functions as shown in Figure 10. Table V shows the required test procedure to locate all faults, except the ones at the outputs $Z_{31}$ and $Z_{32}$, by observing only the outputs of the three AR functions ($Z_{12}$, $Z_{22}$, $Z_{32}$). Outputs $Z_{31}$ and $Z_{32}$ will be inputs to some other LRU's and therefore their faults will be localized by the AR function of those LRU's. Recall that for each AR function four tests are required (Table III). If the

<table>
<thead>
<tr>
<th>Test No.</th>
<th>Test Procedure</th>
<th>Inputs to $F_{13}$</th>
<th>$A$</th>
<th>$B$</th>
<th>$C$</th>
<th>$D$</th>
<th>$E$</th>
<th>$F$</th>
<th>$G$</th>
<th>$Z_{31}$</th>
<th>$Z_{32}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 1 0 1 1 0 0 0</td>
<td>1 1 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1 1 0 1 0 0 0 0</td>
<td>1 1 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1 1 1 0 0 0 0 0</td>
<td>1 1 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1 1 1 1 0 0 0 0</td>
<td>1 1 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1 0 0 1 0 0 0 1</td>
<td>1 1 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1 0 1 0 0 1 0 0</td>
<td>1 1 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1 1 0 0 1 0 0 0</td>
<td>1 1 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0 0 0 0 0 0 0 0</td>
<td>0 1 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0 1 1 1 0 0 0 0</td>
<td>0 0 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0 1 1 0 0 1 1 0</td>
<td>0 1 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1 1 1 1 1 0 1 0</td>
<td>1 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0 1 1 1 1 1 0 0</td>
<td>0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>- - - - 1 1 1 - -</td>
<td>0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 9—The LRU with AR function

\[
Z_1 = \overline{A_1 A_2 A_3} \\
Z_2 = \overline{A_1 + A_2 + A_3} \\
Z_3 = A_1 A_2 + \overline{A_1} A_3
\]

Figure 10—Implementation of the subsystem using LRU's with AR functions

Figure 11—Use of a converging connection to reduce the points which must be made observable

TABLE V—A Test Procedure for the Subsystem of Figure 10 with AR Functions

<table>
<thead>
<tr>
<th>Test No.</th>
<th>Test Procedure</th>
<th>Inputs to F₂₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Conditions corresponding to this test pattern are applied to each of the AR functions, then the fault localization will be achieved. The first four tests provide the conditions for the test pattern of the AR functions in \(F_{21}\) and \(F_{22}\). In so doing, by applying appropriate conditions at the input \(A\) we also manage to provide the conditions corresponding to two of the four tests for \(F_{23}\). The last two tests provide the remaining tests for \(F_{23}\). Note that we use the results at the output \(Z_{23}\) only if we have determined that no input fault exists at \(F_{21}\) and \(F_{22}\) i.e., when \(Z_{11}\) and \(Z_{22}\) do not show a failure. Thus the output \(Z_{23}\) is used only to detect and locate the input faults of \(F_{23}\).

Thus a test procedure with six tests and three observable points is sufficient to localize the failures. If we would like to reduce the number of observable points further, we can use a converging (Section 7) AR function \(F_1\) such as the 2-input AR function of an earlier section (Figure 11).³ The points which must be observed are the outputs \(Z_{33}\) and \(Z_{41}\). But when we reduce the number of observable points by converging we pay a penalty in the length of test sequence. Recall that, as per upper bound it would take 11 tests to test the convergent structure (as a two input AR function requires three tests). Two additional tests will be required to provide the remaining tests for the AR function \(F_{23}\) as before. And thus a total of 13 tests may be necessary. However, as shown in the test procedure of Table VI, the actual number of tests required for the converging connection is only eight, and therefore, a total of nine tests are required (Tests 1 and 5 are identical). Note that the required test pattern for the
TABLE VI—A Test Procedure for the Configuration of Figure 11

<table>
<thead>
<tr>
<th>Test No.</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

converging AR function $F_1$ is applied in the process of applying test patterns for the AR functions of $F_{21}$ and $F_{22}$. Table VII summarizes these results for the sub-system considered in the illustration.

It is important to note that the AR function design is unrelated to the function actually performed on an LRU, but is only dependent on the number of inputs. Thus in practical LSI applications, as the level of integration goes up and as the gate/pin ratio increases, the percentage redundant logic required for AR function will decrease.

TABLE VII—Comparisons of the Different Approaches to Implement a Subsystem (Figures 8, 10, and 11)

<table>
<thead>
<tr>
<th>No.</th>
<th>Case</th>
<th>No. of Points Which Must be Made Observable</th>
<th>Length of the Test Sequence for Fault Localization</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LRU's without AR functions (Figure 8)</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>LRU's with AR functions (Figure 10)</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>LRU's with AR functions and a converging AR function (Figure 11)</td>
<td>2</td>
<td>9</td>
</tr>
</tbody>
</table>

CONCLUSIONS

In this paper we have introduced the notion of adding special Ambiguity Resolver functions to Least Replaceable Units in order to provide localization of input/output failures. We have demonstrated the existence of these functions for any number of inputs and shown how they can be synthesized by means of a conformal approach. For those cases in which it is important to further reduce the number of points that must be made observable, a convergent structure is discussed, along with an upper bound on the length of the required test sequence.

The Ambiguity Resolver approach is certainly not the complete answer to the fault localization problem in Large Scale Integrated circuitry, but it is a step in the direction of tailoring the diagnostic technique to the characteristics of the environment.

REFERENCES

1 H Y CHANG  E G MANNING  G METZE  
Fault diagnosis of digital systems
Wiley-Interscience New York 1970 Chapter II

2 G L SCHNABLE  R S KEEN JR  
Failure mechanisms in large-scale integrated circuits

3 J E McCORMICK  
On the reliability of microconnections
Electronic Packaging and Production Vol 8 No 6 June 1968 pp 187-189
4 D B ARMSTRONG  
*On finding a nearly minimal set of fault detection tests for combinational logic nets*

5 J P ROTH  
*Diagnosis of automata failures—A calculus and a method*

6 W H KAUTFZ  
*Fault testing and diagnosis in combinational digital circuits*
IEEE Trans on Computers EC-17 pp 352-366

7 M A MEHTA  
*A method for optimizing the diagnostic resolution of input-output faults in digital circuits*  
PhD Thesis Illinois Institute of Technology 1971