NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.
RELIABILITY MODELING OF COMPENSATING MODULE FAILURES IN MAJORITY VOTED REDUNDANCY

Daniel P. Siewiorek

Department of Computer Science
Carnegie-Mellon University
Pittsburgh, PA. 15213

October 1974

This work was sponsored by the Advanced Research Projects Agency of the Office of the Secretary of Defense under contract F44620-73-C-0074.
ABSTRACT

The classical reliability model for N-modular redundancy (NMR) assumes the network to be failed when a majority of modules which drive the same voter fail. It has long been known that this model is pessimistic since there are instances, termed compensating module failures, where a majority of the modules fail but the network is nonfailed. A different module reliability model based on lead reliability is proposed which has the classical NMR reliability model as a special case. Recent results from the area of test generation are employed to simplify the module reliability calculation under the lead reliability model. First a fault equivalent technique, based on functional equivalence of faults, is developed to determine the effect of compensating module failures on system reliability. This technique can increase the predicted mission time (the time the system is to operate at or above a given reliability) by at least 40% over the classical model prediction for simple networks. Since the fault equivalent technique is too complex for modeling of large circuits a second, computational simpler technique, based on fault dominance, is derived. It is then shown to yield results comparable to the fault equivalent technique. A more complex example circuit analyzed by the fault dominance model shows at least a 75% improvement in mission time due to modeling compensating module failures. A commercially available 31 gate integrated circuit chip is also modeled to demonstrate the applicability of the technique to large circuits.

Key Phrases: Triple modular redundancy (TMR), compensating module failures, fault equivalence, fault dominance, mission time improvement.

INTRODUCTION

New system designs for reliable computers must be explored to meet the increasing demand for reliable computing systems. One important method of predicting the performance of a system is the modeling of the system reliability.

Modeling requires a mathematical or physical representation which incorporates the salient parameters of the modeled system [1]. A model is an incomplete representation of the subject under study. To be of value, the modeling technique must be convenient to apply and must successfully predict the behavior of the subject under various parameter changes. If a reliability model is accurate, then insights can be gained as to how the system reliability changes as a function of the design parameters.

An exact method to model the effect of a majority of failed modules in the N-modular redundancy (NMR) scheme is presented and shown to increase the predicted mission time (the time for which the system is to operate at or above a given reliability) over that of the classical reliability model by at least 40%. This exact method is too complex to apply to a large circuit. Thus a second and
computationally simpler method is developed and shown to yield a predicted mission time within 10% of the exact model for example systems.

CLASSICAL NMR RELIABILITY MODEL

NMR [2] is implemented by dividing the nonredundant network into modules, replicating the modules \( N \) times (where \( N = 2t + 1 \) and \( t \) is an integer), and inserting a majority gate between each set of replicated modules. Figure 1 depicts the implementation of a triple modular redundancy (TMR) version of a portion of a nonredundant network consisting of a two input, single output module. TMR will be the major topic of discussion, although the procedures presented have straightforward applications to the general case of NMR.

Classically the reliability of the network in Figure 1 is modeled by assigning the modules a reliability function, call it \( R_m(t) \), or \( R_m \) with time as an understood variable. The probability of module failure is thus \( 1 - R_m \). It is then assumed that the system fails when two or more modules driving the same voter, say voter \( A \) in Figure 1, fail. The classical reliability model is:

\[
R_m^3 + 3R_m^2(1 - R_m)
\]

(1)

The effect of nonperfect voters can readily be incorporated into (1) if voters are assigned to module inputs [3,4,5]. Since each voter drives exactly one module input, a voter failure has the same effect as a module failure. If \( R_v \) is the voter reliability, then the effective module reliability in (1) becomes \( R_v^2R_m \). Networks that do not have a voter for every module input can be modeled by more complex techniques [6]. However, for the present discussion we will assume every module input has an associated voter. Further, the discussion will center on applying the modeling techniques to modules only. Nonperfect voters can easily be modeled by applying the techniques to modules and their associated input voters.

Equation (1) is pessimistic since there are many cases that a majority of the modules are failed yet the network of Figure 1 would not be failed. For example, consider two failed modules for the network of Figure 1. Assume module one has a permanent logical one on its output while module three has a permanent logical zero output. The network will still realize its designed function. Such multiple module failures which do not lead to network failures will be termed compensating module failures.

Adding these double, and even triple, module failure cases can often lead to a substantially higher predicted reliability than the classical reliability model. With a better reliability model some systems previously designed may be found to be overdesigned for their specific mission because an inadequate reliability model was used.
Figure 1. Classical triple-moduler redundancy.
MODELING COMPENSATING FAILURES

In the literature, equation (1) is sometimes rewritten to take into account the cases where two modules can fail so as to have compensating effects at the voter:

\[ R = R_m^3 + 3R_m^2 (1 - R_m) + K(3 R_m^2 (1 - R_m)^2) \]  (2)

The K in (2) is a probability formed by the ratio of the number of ways in which compensating failures can occur divided by the number of ways any failure can occur. K has often been taken as 1/2 [7].

An alternative model for compensating failures that has appeared in the literature [7] is:

\[ R_{TNR} = R_m^3 + 3R_m^2 (1 - R_m) + \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \sum_{r=0}^{\infty} K_{mnr} \frac{P_{mnr}(1)}{m+n+r} \]  (3)

where the module failures follow the Poisson assumption; there are \( K_{mnr} \) ways of designating which of the three modules have \( m \), \( n \), and \( r \) failures respectively; and \( P_{mnr} \) is the probability that the system operates correctly with \( m \) failures in one module, \( n \) failures in another, and \( r \) failures in the other.

Equation (3) can be rewritten as

\[ R_{TNR} = R_m^3 + 3R_m^2 (1 - R_m) + R_{Two} + R_{Three} \]  (4)

where \( R_{Two} \) and \( R_{Three} \) is the contribution to the system reliability from compensating failures in two and three modules respectively.

Methods for calculating \( K \) or \( P_{mnr} \) are not described in the literature. The next two sections will develop a technique to calculate \( R_{Two} \) and \( R_{Three} \) based on a lead failure model for module reliability. The technique can also be used to calculate the \( P_{mnr} \) of equation (3), if some assumptions about the relationship between the lead failure model and Poisson failure model of module reliability are made.

This exact technique is only practical for small circuits. A computationally simpler method, employing the concepts of fault equivalence and fault dominance, is derived for determining the contribution to \( R_{Two} \) of two failed modules with one failure in each. The latter method is validated as a good approximation by comparison to the exact technique.

MODULE FAILURE MODEL

In order to calculate \( R_{Two} \) and \( R_{Three} \) we will have to define what we mean by a module failure. Research in the area of testing and diagnosing combinational and sequential logic circuitry has relied heavily on the logical stuck-at-fault model [8]. This model assumes that most or all failures of interest in a logic circuit manifest themselves as some line in the circuit taking on a constant logical
value, either one or zero. Now that algebraic structure which applies to the behavior of networks in
the presence of stuck-at faults has been developed [8,11,12], the tools are available to formulate and
analyze a new module reliability model.

The new model will assign a reliability function to each lead in the network rather than each mod­
ule as in the classical model. Lead reliability will be represented by R and the probability of lead
failure by 1 - R.

Much has been written in defense of the stuck-at failure model [8] but a few words will now be de­
voted to justification of the lead reliability model. In one study of IC failure mechanisms [9] it was
found that about 84% of the IC failures were directly related to lead failures, either of input leads or
of metalization on the chip itself.

Similar to the classical model assumption that module failures are statistically independent events,
it will also be assumed that lead failures are statistically independent. A further advantage of the
lead reliability model is that it takes into account the increased number of interconnections required
for the massive redundancy version of a nonredundant system. Wiring errors and off-chip interconnec­
tions than may be the major source of failures.

It will be assumed that the stuck-at-one (s-a-1) faults are as likely to occur as the stuck-at-zero
(s-a-0) faults. Hence, the probability that a lead is s-a-1 is:

\[ P(\text{s-a-1}) = P(\text{lead failure}) \cdot P(\text{s-a-1 | lead failure}) \]

\[ P(\text{s-a-1}) = (1-R) \cdot \frac{1}{2} \]

FAULT EQUIVALENT RELIABILITY MODEL

Using the above model for module failure it is now possible to calculate \( R_{\text{two}} \). First, a few assump­
tions are necessary. The modules will be assumed to consist of irredundant combinational logic [8] so
that any single internal module fault will cause an improper output for at least one set of inputs. It
will also be assumed that the system has failed as soon as it is possible for the voter to give a wrong
response to any possible input combination. This excludes the situations where a module trio fails but
subsequent faults within the module trio restores proper behavior.

To model the faulty modules we will adopt the notation developed in [8]. We will now illustrate
the evaluation of the \( R_{\text{two}} \), i.e., the case of two faulty modules for a simple module.

1) Transform the logical circuit into the corresponding logical model [8].

Consider Figure 2(a) where the module under study is a single two input NAND gate. The logical
model is a directed graph shown in Figure 2(b).

2) Form the functional equivalence classes for all single and multiple faults in the logical
Fig. 2. An example module (a) for the calculation of supplementary classes and (b) its logical model.
model [8].

A fault is said to be functionally equivalent to another fault if and only if the output function realized by the module with only the first fault present is equal to the function realized when only the second fault is present. For example, a/0 and c/1 (the notation l/i means line l stuck at logical value i) are functionally equivalent. Table 1(a) shows the fault classes and their members. Here 0 is the null fault and represents the fault free network. The functional equivalence classes are assigned numbers arbitrarily.

For a TMR system, two faults, \(f_1\) and \(f_2\), occurring in different modules are said to be supplementary if their simultaneous presence does not cause network failure. Two functionally equivalent fault classes are called supplementary classes if the faults contained in one class are supplementary to faults in the other.

3) Enumerate the supplementary classes.

For the case of two module failures one of the modules will be the fault free function.

The majority gate can be considered to be a threshold gate with input weights 1 and threshold of 2 [10].

In Table 1(b) the Karnaugh maps represent the fault functions for the faults \(x, a/1,\) and \(b/1\) respectively. We continue to try all possible combinations of faulty function pairs until all supplementary classes are formed. These are shown in Table 1(c) for our example.

In the last step a matrix \(E\) is used to actually evaluate \(R_{Two}\). Element \(E_{i,j}\) of equivalence class matrix \(E\) is the number of faults in equivalence class \(j\) (the equivalence classes were assigned numbers under step 2) which are a result of \(i\) leads in a module failing, where \(i\) is termed the fault multiplicity.

4) Form the term for two faulty modules by use of the equivalence class matrix \(E\) and the equation:

\[
R_{Two} = \left(3 \sum_{k=2}^{2p} \left(1/2^k \sum_{j=1}^{k-1} (E_{i,j} \cdot E_{k-j,j}) R^{3p-k}(1-R)^k \right) \right)
\]

\(\forall i,j\) such that \((i,j)\) is a supplementary class

where \(p\) is the number of leads in a module and \(k\) is the sum of the line failures in the two failed modules.

The development of step 4) is best given by an example. The equivalence class matrix for the NAND gate of Figure 2 is derived from the entries in Table 1(a) and is shown in Table 2.

For our example of two failed NAND modules (6) becomes:

\[
\binom{3}{2}[(20/4)R^7(1-R)^2 + (72/8)R^6(1-R)^3 + (118/16)R^5(1-R)^4 + (96/32)R^6(1-R)^5 + (32/64)R^3(1-R)^6]
\]
Table 1. The (a) fault classes, (b) an example of the test for supplementary fault classes, and (c) the supplementary classes for the NAND gate example of Fig. 2.

| Class   | Fault Function | Maps |
|---------|----------------|------|
| $C_{F1}$ : {λ} | $\bar{X} + \bar{Y}$ | $\begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 1 & 1 \\ \hline \end{array}$ |
| $C_{F2} = \{a/1\}$ | $\bar{Y}$ | $\begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array}$ |
| $C_{F3} = \{b/1\}$ | $\bar{X}$ | $\begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array}$ |
| $C_{F4} = \{c/0; a/1, c/0; a/0, c/0; a/1, b/1; b/1, c/0; b/0, c/0; a/1, b/1, c/0; a/1, b/0; c/0; a/1, b/0, c/0; a/0, b/1, c/0\}$ | \[ \] | $\begin{array}{c|c|c} X \oplus Y & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array}$ |
| $C_{F5} = \{a/0; b/0; c/1; a/1, c/1; a/0, c/1; b/1, c/1; b/0, c/1; a/1, b/0; a/0, b/1; a/1, b/1, c/1; a/1, b/0, c/1; a/0, b/1, c/1\}$ | \[ \] | $\begin{array}{c|c|c} X \oplus Y & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array}$ |

(b) Threshold Voter Output Function

\[
\begin{array}{ccc}
\lambda & \text{Threshold Map} & \text{Voter Output Function} \\
#1 & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 1 & 2 \\ \hline \end{array} & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array} \\
#2 & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 1 & 1 \\ \hline \end{array} & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array} \\
#3 & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 1 & 1 \\ \hline \end{array} & \begin{array}{c|c|c} X & 0 & 1 \\ \hline Y & 0 & 1 \\ \hline \end{array} \\
\end{array}
\]

\begin{align*}
(2,3) & \quad (2,5) & \quad (3,5) & \quad (4,5) \\
(3,2) & \quad (5,2) & \quad (5,3) & \quad (5,4)
\end{align*}
Table 2. The equivalence class matrix for the NAND gate of Figure 2.

| Number of Failed Leads | 0 | 1 | 2 | 3 | 4 | 5 |
|------------------------|---|---|---|---|---|---|
| 0                      | 1 | 0 | 0 | 0 | 0 | 0 |
| 1                      | 0 | 1 | 1 | 1 | 3 | 3 |
| 2                      | 0 | 0 | 0 | 5 | 7 | 7 |
| 3                      | 0 | 0 | 0 | 4 | 4 | 4 |

The procedure outlined above is easily modifiable to handle the case of three module failures and is readily extendable to other multiple line redundancy schemes (NMR). In some logic families one type of logical stuck-at fault may be much more likely than the other. If so, P(s-a-i | lead failure) could be taken as approximately 1.0 and the fault equivalence classes formed by considering s-a-i type faults only. The comparison of this reliability model with the one of equation (1) will now be undertaken.

COMPARISON OF FAULT EQUIVALENT AND CLASSICAL RELIABILITY MODELS

If there are p leads in a module, then the module reliability, \( R_m \), according to the fault equivalent model just presented is \( R^p \). For the case of fewer than half the modules failing in an NMR network, the classical reliability model gives a reliability of:

\[
R = \sum_{i=0}^{[N/2]} \binom{N}{i} R_m^{N-i} (1 - R_m)^i \tag{8}
\]

It will now be shown that the first \([N/2] + 1\) terms of the fault equivalent reliability model are identical to (8), the classical NMR reliability model.

Theorem 1: The fault equivalent reliability model proposed above has the same form as (8) for \([N/2]\) or fewer module failures.

Proof: The probability of no module failures is \( (R^p)^N = R_m^N \) which is the first term of (8). Now for any number of module failures less than or equal to \([N/2]\), there is still a majority of working modules and any failure configuration of a failed module's lines would not cause system failure. So to complete the proof of the theorem, all we need show is that the classical failure probability \((1-R_m)\) is equal to the single module failure probability of the fault equivalent model.

The fault equivalent failure probability for a single module is:

\[
\sum_{k=1}^{p} \frac{1}{2^k} \cdot \sum_{i=1}^{p-k} R^p \cdot R_m^{k-1} (1-R_m)^k \tag{9}
\]
The $\kappa_{k,1}$ term, considering the cases for all $i$, is $2^P \binom{P}{k}$ since there are $\binom{P}{k}$ ways to select $k$ failed leads from $p$. Each failed lead may be in one of two failure modes, $s-a-1$ or $s-a-0$, which accounts for the $2^k$. Hence (9) becomes:

$$\sum_{k=1}^{P} \binom{P}{k} (1-R)^k R^{P-k} = R^P + 1$$

which is the probability of module failure using the classical reliability model. This completes the proof of Theorem 1.

The classical reliability model (1) was compared to the fault equivalent model for the single NAND gate module of Figure 2 and a two NAND gate module. The comparison was made in terms of mission time improvement $I$ [7]. $I$ is the ratio of the time at which two reliability models have the same reliability. To obtain $I$, the classical reliability at time $t$ was equated to the new reliability at time $t'$ and the resultant equation solved for $I$. The mission time improvement for the fault equivalent model (evaluated for compensating failures in only two modules) over the classical reliability model is shown in Table 3. It is important to note that the apparent improvement in mission time is not due to any change in the modeled hardware system but rather to a more accurate reliability model.

Table 3. Mission time improvement, $I$, of $[R_m^3 + 3 R_m^2 (1-R_m) + R_{\text{two}}]$ over $[R_m^3 + 3 R_m^2 (1-R_m)]$ for two simple modules.

|        | 0.7 | 0.8 | 0.9 | 0.95 | 0.99 |
|--------|-----|-----|-----|------|------|
| Single NAND gate | 1.474 | 1.477 | 1.484 | 1.491 | 1.496 |
| Two NAND gate      | 1.491 | 1.497 | 1.515 | 1.526 | 1.539 |

It can be seen that a mission time improvement of 50% can be obtained by adding the effect of $R_{\text{two}}$ to the classical reliability model. Another way of looking at the parameter $I$ is that if the classical model is used, then the resultant system is overdesigned by 50% since it could meet its mission time specification with less reliable components.

The technique outlined above for the fault equivalent reliability model can also be employed to determine the $P_{mn}$ term of equation (3). In (3) a module is assumed to follow the Poisson distribution, e.g. the probability that there are exactly $n$ failures in a given period of time $t$ is:

$$R_m \frac{(\lambda t)^n}{n!}$$

In (3) a module can have an infinite number of failures. If we associate a failure in a module to a lead
failure and assume \( P_{mn} = 0 \) for \( m, n, r > p \), where \( p \) is the number of leads per module, then \( P_{mn} \) is given by:

\[
\frac{1}{(m-1)(n-1)p^{m+n}} \left( E_{mi} \cdot E_{nj} \right) \tag{11}
\]

The first term is the number of ways two modules can fail with \( m \) failures in one and \( n \) in the other. Further, each lead can fail in one of two ways \((s-a-0, s-a-1)\). The second term of (11) is the number of ways two modules with \( m \) and \( n \) lead failures can form compensating failures. The modifications of (11) to calculate \( P_{mn} \) are obvious.

The fault equivalence class matrix \( E \) is the computational bottleneck for the fault equivalent model. A significantly simpler model, based on fault dominance, is developed in the next section. Subsequently it will be compared to the fault equivalent model and shown to be a tight lower bound.

**FAULT DOMINANCE RELIABILITY MODEL**

One might speculate that the first term in the summations of equation (3) \((m-1, n-1, r = 0)\) or equation (6) \((k = 2)\) might be the dominant term as far as compensating failures are concerned. That this is the case will be demonstrated by comparison with the fault equivalent model and a multiple fault model. Hence we shall consider the case of two module failures with one failure per module. A simple formula for \( P_{110} \) and \( R_{220} \) (with one lead failure in each of two modules) assuming tree structured modules will now be derived. Approximations for modules with reconvergent fanout will also be given.

The modules adhere to the same assumptions as the fault equivalent reliability model. The following definitions will be required:

**Definition 1:** A test set, \( T_i \), for a fault \( f_i \) is that set of inputs which can detect fault \( f_i \); i.e., produces a different output if \( f_i \) occurs than when no fault is present.

**Definition 2:** Fault \( f_2 \) is equivalent to fault \( f_1 \), denoted by \( f_1 = f_2 \), if and only if \( T_1 = T_2 \), where \( "=" \) denotes set equivalence.

**Definition 2:** Fault \( f_2 \) is said to dominate \( f_1 \), denoted by \( f_2 > f_1 \) iff \( T_2 \supset T_1 \) where \( \supset \) denotes set covering. Conversely \( f_1 \) is said to be dominated by \( f_2 \).

The necessary and sufficient conditions for compensating module failures can now be given. \( \emptyset \) stands for the empty set.

**Theorem 2:** Single faults \( f_1 \) and \( f_2 \) are supplementary (written \( f_1 \sim f_2 \)) if and only if \( T_1 \cap T_2 = \emptyset \).
Proof:

Part I: If $T_1 \cap T_2 = \emptyset$ then $f_1 \sim f_2$.

For any input $t_1 \in T_1$, $F(f_1) \oplus F = 1$ where $F$ is the nonfaulty output and $F(f_1)$ is the output with fault $f_1$ present. But $F(f_1) \oplus F = 0$ or $F(f_2) = F$ since $t_1$ does not detect $f_2$. Thus majority $(F(f_1), F(f_2), F) = F$. Similarly for any input $t_2 \in T_2$, $F(f_2) \oplus F = 1$ and $F(f_1) \oplus F = 0$. So majority $(F(f_1), F(f_2), F) = F$. Hence if $T_1 \cap T_2 = \emptyset$ then $f_1 \sim f_2$.

Part II: If $f_1 \sim f_2$ then $T_1 \lor T_2 = \emptyset$.

The proof is by contradiction. Assume $f_1 \sim f_2$ but $T_1 \lor T_2 = \emptyset$. Take the case $T_1 \lor T_2 = t_1$. Since $t_1$ is a test for $f_1$, $F(f_1) \oplus F = 1$ for the input $t_1$. Likewise $t_2$ is a test for $F_2$ and $F(f_2) \oplus F = 1$. The majority $(F(f_1), F(f_2), F) \neq F$ and $f_1 \neq f_2$. The original assumption is contradicted and therefore if $f_1 \sim f_2$ then $T_1 \lor T_2 = \emptyset$. This completes the proof.

There are two corollaries to Theorem 2 which will prove helpful.

Corollary 2.1: If $f_1 = f_2$ then $f_1 \neq f_2$.

Proof: If $f_1 = f_2$ then $T_1 = T_2$ by the definition of fault equivalence. Thus $T_1 \lor T_2 \neq \emptyset$ and $f_1 \neq f_2$.

Corollary 2.2: If $f_1 > f_2$ then $f_1 \neq f_2$.

Proof: If $f_1 > f_2$ then $T_1 \supset T_2$ by the definition of fault dominance. Further $T_2 \neq \emptyset$ since the module was assumed redundant and a fault for which there is no detection test is considered redundant. Hence $T_1 \lor T_2 \neq \emptyset$ and $f_1 \neq f_2$.

The modules will also be assumed to be composed of elementary gates: Invert, AND, OR, NAND, and NOR are depicted in Figure 3. The equivalent and dominating fault class structure, as developed in [8, 11, 12], is also depicted. Consider the two input AND gate in Figure 3. In the graphical representation of the AND gate each lead is represented by a pair of circles; the upper circle stands for a s-a-1 fault, the lower one for a s-a-0. Equivalent faults are connected by a straight line. Faults related by dominance are connected by an arrow pointing from the dominating fault to the dominated fault. The test sets for the various faults are also given in Figure 3. The notation $T_{a/1}$ means the test set for input 'a' being s-a-1. The first element of the test vector corresponds to the uppermost lead, etc. Thus, to detect a s-a-1 in the two input AND gate a 0 should be applied to input 'a' and a 1 to input 'b'.

An important observation to make from Figure 3 is that for elementary gates the test sets for all faults, other than equivalent faults, are disjoint. This includes faults dominated by the same output fault. A test for a circuit can only put one value on each lead of a circuit, thus the circuit test sets for two faults in an elementary gate must be disjoint if the faults are not equivalent.
Figure 3. The elementary gates (a), the test sets (b) and the class structure (c)
A tree circuit is depicted in Figure 4(a) and its fault class structure in Figure 4(b). Although AND and OR gates are used the fault class structure of Figure 4(b) is representative of the fault class structure of any arbitrary tree composed of elementary gates.

The exact number of supplementary faults for a given lead can be derived with the aid of the following definitions.

**Definition 4**: A lead y is a successor of a lead x iff every path from x to the circuit output also passes through y.

**Definition 5**: A lead x is a predecessor of a lead y iff y is a successor of x.

In Figure 4(a) lead y is a successor of lead x and x is a predecessor of y.

**Theorem 3**: The number of supplementary single faults to a failure on lead x is an arbitrary tree circuit composed of arbitrary combinations of elementary gates is:

\[ p + p - n_{\text{pre}} - n_{\text{suc}} \]  

(12)

where \( p \) is the number of leads, \( n_{\text{pre}} \) the number of predecessor leads of lead x and \( n_{\text{suc}} \) the number of successor leads of x.

**Proof**: For a tree with \( p \) leads the fault class structure will always consist of two trees of \( p \) nodes each regardless of what combination of elementary gates are used. Of the \( 2p \) single faults, any fault in one tree is supplementary with any fault in the other tree since their test sets are disjoint. A lower bound on the number of supplementary faults is thus \( p \). To illustrate this consider faults \( f_4 \) and \( f_6 \) in Figure 4. If we denote their test sets by \( T_4 \) and \( T_6 \), respectively, it is easy to demonstrate that \( T_4 \cap T_6 = \emptyset \). No matter what the relative positions of \( f_4 \) and \( f_6 \) there is always a fault in one tree such that \( T_8 = T_4 \cup p \) (where \( \cup \) denotes set union) and a corresponding fault in the other tree such that \( T_h = T_6 \cup r \). Also the faults \( f_8 \) and \( f_h \) will be in the same elementary gate so that \( T_8 \cap T_h = \emptyset \). Thus

\[ (T_4 \cup p) \cap (T_6 \cup r) = \emptyset \]
\[ (T_4 \cap T_6) \cup (T_4 \cap r) \cup (p \cap T_6) \cup (p \cap r) = \emptyset \]

This can be the empty set only if each individual term is empty. Hence \( T_4 \cap T_6 = \emptyset \).

Within each tree of the fault class structure any fault on lead x will either dominate or be equivalent to any predecessor node because of the transitivity of the dominance and equivalence relationships [11,12]. Hence there can be no supplementary failures on the \( n_{\text{pre}} \) predecessor leads since a test set for a fault on a predecessor lead will not be disjoint from the test set for a fault on lead x. Similarly, faults on successor leads to x cannot be supplementary to x since they will either dominate or be
Figure 4. An example tree circuit (a) and its fault class structure (b).
Proof: From the first term of (12) each faulty lead is supplementary to p other lead faults in the other module. Since there are p leads in a module and two types of failures per lead the first term in (13) should represent 2p. We must show that p = \frac{t^{b+1} - 1}{t - 1}. At each level i of the tree there are t leads. Thus p = \sum_{i=0}^{b+1} t^i - \frac{t^{b+1} - 1}{t - 1}. The remaining terms count the number of leads that are not predecessors or successors of a given lead, for all leads. For level k there are t_k leads which are not predecessors or successors to subtrees consisting of \ell-k, \ell-k+1, ..., \ell-1 levels. Levels are numbered from the module output to input. Thus the number of supplementary faults is augmented by:
which is:

$$\sum_{k=1}^{L} t^k \sum_{j=t-k+1}^{L} \frac{t^{j-1}}{t-1}$$

Equation (14) becomes:

$$\sum_{k=1}^{L} \frac{1}{(t-1)} t^k \left( \frac{t^{j+1} - t^{k+1}}{t-1} \right)$$

(15)

$$\sum_{k=1}^{L} \left( \frac{t^{j+1}}{(t-1)} \left[ \frac{t^{j+1} - t}{(t-1)} \right] - \frac{t}{(t-1)} t^{k+1} - \sum_{k=1}^{L} k t^k \right)$$

(16)

It remains to be shown that

$$\sum_{k=1}^{L} k t^k = - \frac{1}{(t-1)} \left( \frac{2}{t-1} (t^{j+1} - t) + t - (t^{j+1} - t) \right)$$

$$\sum_{k=1}^{L} k t^k = t \sum_{k=2}^{L} k t^{k-1} + t = t \sum_{k=1}^{L} (k+1) t^k + t$$

$$= t \sum_{k=1}^{L} (k+1) t^k + t - (t^{j+1} - t) t^{k+1}$$

$$= \sum_{k=1}^{L} k t^k + t \sum_{k=1}^{L} t^k + t - (t^{j+1} - t) t^{k+1}$$

$$= - \frac{1}{(t-1)} \left( \frac{1}{t-1} (t^{j+1} - t) + t - (t^{j+1} - t) t^{k+1} \right)$$

This completes the proof of the theorem.

The subtree counting technique employed in Theorem 4 can be used to calculate the number of supplementary faults, \(S_2\) (for a single lead failure in each module) for an arbitrary tree. Thus, from equation (6)

$$R_{two} = 3 S_2 \cdot \frac{(1/2)^2}{K^{3p-2}} \frac{(1-R)^2}{(1-R)^2}$$

(17)

Note that for the NAND gate of Figure 2 \(K = 1\) and (13) yields \(S_2 = 20\). Thence (17) agrees with the first term of equation (7). Similar correspondences have been made between the fault equivalence and fault dominance models for tree structured modules. Table 4 depicts the mission time improvement of the fault dominance reliability model over the classical reliability model for various circuits. A four level binary tree is also listed in Table 4 to demonstrate that the fault dominance model can lead to
substantial mission time improvement.

### Table 4. Mission time improvement, \( I \), of the Fault equivalence reliability model and Fault dominance reliability model over the classical reliability model for various modules.

|             | \( R \) |
|-------------|---------|
|             | 0.75    | 0.85   | 0.9    | 0.95   | 0.99   |
| **Single NAND gate** |         |         |         |         |        |
| Equivalence Model | 1.476   | 1.477  | 1.481  | 1.491  | 1.496  |
| Dominance Model   | 1.358   | 1.382  | 1.405  | 1.439  | 1.472  | 1.491  |
| **Two NAND gate** |         |         |         |         |        |
| Equivalence Model | 1.494   | 1.497  | 1.510  | 1.515  | 1.526  | 1.539  |
| Dominance Model   | 1.355   | 1.384  | 1.414  | 1.452  | 1.492  | 1.531  |
| **Four Level Full Nary Tree** |         |         |         |         |        |
| Dominance Model   | 1.405   | 1.451  | 1.505  | 1.575  | 1.663  | 1.766  |
| Multiple Fault Model | 1.300  | 1.318  | 1.389  | 1.361  | 1.386  | 1.408  |
| Dominance plus Multiple | 1.442  | 1.485  | 1.535  | 1.598  | 1.692  | 1.771  |

In order to illustrate that equation (13) is the dominant term in equation (6) a reliability model employing multiple faults can be developed using the following theorem:

**Theorem 5**: A lower bound on the number of supplementary faults for \( N \) lead failures in two modules of a full \( L \) level tree of \( t \)-input elementary gates is:

\[
S_N = 2^{N-1} \sum_{i=1}^{N-1} \left\{ \sum_{k=0}^{L} \binom{L}{k} (t^{L-k+1} - 1) \right\} \left\{ \sum_{k=0}^{L} \binom{L}{k} (t^{L-k+1} - 1) \right\}
\]

**Proof**: Equation (18) enumerates the **subtree multiple faults** (SMF), those multiple faults whose component faults are all in one subtree with one fault at the root of the subtree. Since the fault at the root masks the effects of other faults in the subtree then the SMF (subtree multiple fault) will be supplementary to any other SMF in the other module so long as the root faults are in different fault class structure trees. At level \( k \) there are \( t^k \) subtrees each with \( t^{L-k+1} \) branches. Thus there are \( \sum_{k=0}^{L} \binom{L}{k} (t^{L-k+1} - 1) \) SMFs where \( \binom{x}{y} \) is defined as zero if \( y > x \) or \( y < 0 \). The outer sum in (18) enumerates the ways \( N \) faults can be distributed between two modules with at least one fault per module. Finally, only two of the \( N \) faults have specified values (those at the roots of the subtrees) hence there are \( 2^{N-2} \) values of \( s-a-1 \) or \( s-a-0 \) the multiple fault can assume and still be supplementary. Finally, there are
two ways the root faults can be in different fault class structure trees. This accounts for the $2^{n-1} = 2 \cdot 2^{N-2}$ factor in (18) and completes the proof.

Equation (17) can now be amended to

$$R_{\text{two}} = \frac{2^p}{3} \sum_{N=2}^{p} S_N \left( \frac{1}{2} \right)^N R^{3p-N} \left(1-R\right)^N$$  (19)

Equation (19) is a lower bound since there are multiple supplementary faults which do not have all of their component faults restricted to a single subtree. In Table 4 the Multiple Fault Model is compared to the Dominance Model for a four level binary tree for $S_2$ exact (13), $S_2$ estimated (18) for all $N$, and a combination of double and multiple faults (19). From the comparison we can see that $S_2$ exact is the dominant contributor to the mission time improvement and that multiple faults as given by (18) are a second order effect.

Equation (13) can be used to estimate $S_2$ for arbitrary trees or circuits with reconvergent fan-out. An upper-bound on $S_2$ for an arbitrary tree would be equation (13) with the maximum depth of the tree substituted for $i$ and the maximum number of inputs per gate for $t$. This accounts for all supplementary faults in the circuit and for some that do not exist. A lower bound would be (13) with minimum $i$ and minimum $t$.

The fault dominance model can also be used for circuits with reconvergent fan-out. The circuit is modeled by an identical circuit with fan-out points removed. Figure 5(a) depicts an exclusive-OR circuit and Figure 5(b) the tree circuit used to calculate the supplementary faults. This value for $S_2$ will be a lower bound since not all faults are considered. All supplementary faults in the reduced circuit will also be supplementary in the original circuit since fan-out points only restrict the values of test sets. Gate inputs are no longer independent. Table 5 depicts the mission time improvement for the exclusive-OR and a SN74147 10-line-to-4-line priority encoder chip. The latter consisted of 31 gates and 70 leads. The value of $S_2$ was calculated by inspection. The number of non-successor and non-predecessor leads for each lead, taken over all leads, took less than 10 minutes to determine by hand. This illustrates the applicability of the technique to large circuits.

| Table 5. Mission time improvement, $I$, of the fault dominance reliability model over the classical reliability model for modules with reconvergent fan-out. |
|---|---|---|---|---|---|---|
| $I$ | $R_m$ | 0.75 | 0.8 | 0.85 | 0.9 | 0.95 | 0.99 |
| exclusive-OR | 1.196 | 1.207 | 1.219 | 1.232 | 1.246 | 1.259 |
| priority encoder | 1.228 | 1.244 | 1.263 | 1.283 | 1.304 | 1.324 |
Figure 5. An exclusive-OR circuit (a) and the modeled circuit with fan-out removed (b).
The fault dominance model can also be used to calculate $P_{110}$ by dividing the number of supplementary faults by the total number of single faults. To find the maximum value of $P_{110}$ for a full tree divide equation (13) by $\frac{2^{n+1} - 2}{t-1}$ and take the limit as $t$ approaches infinity. The result is:

$$\frac{1}{2} + \frac{1}{4} \frac{1}{t-1}$$

Hence $P_{110}$ is between $1/2$ and $3/4$ for a binary tree. It can be shown that adding levels to a tree causes $P_{110}$ to increase while adding an inverter or a new gate input causes $P_{110}$ to decrease.

CONCLUSION

A technique, the fault equivalence reliability model, has been developed and shown to increase mission time by 50% for some simple circuits. A computationally simpler model, the fault dominance reliability model, has been demonstrated to be a good approximation to the fault equivalence model. Both techniques can be applied to calculate the $P_{\max}$'s for modules employing the Poisson failure assumption. The classical reliability model may be sufficient for establishing the better of two different fault tolerant architectures where the mission time improvement may be on the order of 5-20. However, in fine tuning of the design and predictions of the reliability of the final system a method which accounts for compensating module failures should be used.

REFERENCES

[1] Bouricius, W. G., W. C. Carter and P. R. Schneider, "Reliability modeling techniques for self-repairing computer systems," Proc. 24 Natl. Conference ACM, Publication P-69, pp. 295-309, 1969.

[2] Mathur, F. P. and A. Avizienis, "Reliability analysis and architecture of a hybrid redundant digital system: generalized triple modular redundancy with self-repair," in 1970 Spring Joint Comp. Conf., AFIPS Conf. Proc., Vol. 36, Washington, D. C.: Thompson, 1970, pp. 375-383.

[3] von Neumann, J., "Probabilistic logics and the synthesis of reliable organisms from unreliable components," Automata Studies, from Annals of Mathematics Studies, No. 34, Princeton University Press, pp. 43-49, 1956.

[4] Brown, W. G., J. Tierney, and R. Wasserman, "Improvement of electronic computer reliability through the use of redundancy," IRE Transactions on Electronic Computers, Vol. EC-10, pp. 407-416, Sept. 1961.

[5] Twoste, R., "Design of a repairable redundant computer," IRE Transactions on Electronic Computers, Vol. EC-11, pp. 643-649, Oct. 1962.

[6] Abraham, J. A. and D. P. Siewiorek, "An algorithm for the accurate reliability evaluation of TMR networks," IEEE Transactions on Computers, Vol. C-23, July 1974.

[7] Bouricius, W. G., W. C. Carter, D. C. Jessop, P. R. Schneider and A. B. Wadia, "Reliability modeling for fault tolerant computers," IEEE Transactions on Computers, Vol. C-20, pp. 1306-1311, Nov. 1971.

[8] McCluskey, E. J. and F. W. Clegg, "Fault equivalence in combinational logic networks," IEEE Transactions on Computers, Vol. C-20, pp. 1286-1293, Nov. 1971.
[9] Platz, E. F., "Solid logic technology computer circuits - billion hour reliability data," in Microelectronics and Reliability, Vol. 8, Great Britain: Pergamon Press, 1969, pp. 55-59.

[10] Hampel, D. and R. O. Winder, "Threshold logic," Spectrum, pp. 32-39, May 1971.

[11] Schertz, D. R. and G. Metze, "A new representation for faults in combinational digital circuits," IEEE Transactions on Computers, Vol. C-21, No. 8, pp. 858-866, Aug. 1972.

[12] Mei, K. C. Y., "Fault dominance in combinational circuits," Technical Note No. 2, Digital Systems Laboratory, Stanford University, Aug. 1970.