R. E. Lyons
W. Vanderkulk

# The Use of Triple-Modular Redundancy to Improve Computer Reliability

**Abstract: One of the proposed techniques for meeting the severe reliability requirements inherent in certain future computer applications is described. This technique involves the use of triple-modular redundancy, which is essentially the use of the two-out-of-three voting concept at a low level. Effects of imperfect voting circuitry and of various interconnections of logical elements are assessed. A hypothetical triple-modular redundant computer is subjected to a Monte Carlo program on the IBM 704, which simulates component failures. Reliability is thereby determined and compared with reliability obtained by analytical calculations based on simplifying assumptions.**

## Introduction

For some time it has been known that the reliability of digital systems can be improved through the use of redundant components, if these additional components are properly employed.[1,2,3] In this paper, we will examine a specific means, called triple-modular redundancy (TMR), for meeting future reliability requirements for digital computers in space and certain military applications. This paper is concerned with system failures caused by permanent component failures, in contrast to the problem of transient failures caused by noise, which has been treated extensively by Von Neumann and others. The same techniques are useful for combating both types of failures.

It is interesting to specify numerically the desired reliability improvement. (Reliability is quantitatively defined as the probability that a system will not fail under specified conditions.) A typical application may require 95-percent reliability for a period of time roughly equal to the mean-time-to-failure of present systems—say one hundred hours. A rough calculation shows that without the use of redundancy this requirement implies a twenty-fold improvement in the mean-time-to-failure of all components. Even if such large improvements in component reliability could be achieved in the years ahead, complex digital systems would still not be reliable enough for those applications where maintenance during operation is impractical. The application of redundancy, together with the improvement of component reliability and the reduction of system complexity, will be required to solve the problem.

The use of redundancy is proposed not as a replacement, but rather as a supplement to the two cardinal principles of reliable design: 1) use the most reliable components and 2) use the least possible complexity consistent with required system performance. This is not just a matter of "using every available means." The analysis shows that the effectiveness of redundancy as a tool for obtaining digital system reliability is much more pronounced in a system composed of basically reliable components than in a system of unreliable components. Put another way, while redundancy can be used as a lever to greatly enhance the reliability of an already reliable system, it is of little use—and can even have a detrimental effect—if the nonredundant system is unreliable in the first place.

The use of redundancy to obtain reliability has been extensively covered in the literature.[4-6] We will attempt to assess the effects on redundant computer reliability due to imperfect voting circuitry and due to the interconnections of logical elements which arise in practice.

## Mathematical analysis of a TMR computer

### *Triple redundancy with perfect voting circuits*

To explain triple-modular redundancy, it is first necessary to explain the concept of triple redundancy as originally envisaged by Von Neumann.[1] The concept is illustrated in Fig. 1, where the three boxes labeled *M* are identical *modules* or *black boxes* which have a single output and contain digital equipment. (A black

box may be a complete computer, or it may be a much less complex unit—for example an adder or a gate.) The circle labeled $V$ is called a *majority organ* by Von Neumann. In this paper it will be called a *voting* circuit because it accepts the input from the three sources and delivers the *majority opinion* as an output. Since the outputs of the $M$'s are binary and the number of inputs is odd, there is bound to be an unambiguous majority opinion.

The reliability of the redundant system illustrated in Fig. 1 is now determined as a function of the reliability of one module, $R_M$, assuming the voting circuit does not fail. The redundant system will not fail if none of the three modules fails, or if exactly one of the three modules fails. It is assumed that the failures of the three modules are independent. Since the two events are mutually exclusive, the reliability $R$ of the redundant system is equal to the sum of the probabilities of these two events. Hence,

$$R = {R_M}^3 + 3{R_M}^2(1 - R_M) = 3{R_M}^2 - 2{R_M}^3 . \qquad (1)$$

Several observations can be made regarding Eq. (1). Note that application of this type of redundancy does not increase the reliability if $R_M$ is less than 0.5. This is an example of the general truth that reliability, even by the use of redundancy, cannot be obtained if the redundancy is applied at a level where the nonredundant reliability is very low. The closer $R_M$ is to unity, the more advantageous the redundancy becomes. In particular, the slope of the curve for the redundant case is zero at $R_M = 1$. Thus, when $R_M$ is very near unity, $R$ departs from unity only by a second-order effect.

Although most of the analysis which follows is valid for any type of dependency of the nonredundant reliability on operating time, it is interesting to examine the specific case where the nonredundant reliability* is a decaying exponential of the operating time, i.e., where

$$R_M(t) = \exp(-ft) = \exp(-t/MTF) . \qquad (2)$$

In this formula $f$ is a constant, called failure rate; and $MTF$ is its reciprocal, called mean-time-to-failure. The reliability of the triply redundant system is now given by

$$R(t) = 3\exp(-2t/MTF) - 2\exp(-3t/MTF) . \qquad (3)$$

Note that for $t > MTF$, which is the range of time that is pertinent to the subject matter of this paper, $R < R_M$. This means that triple redundancy at the computer level should not be used to improve reliability in this case. To obtain improvement in reliability by the use of triple redundancy, we require $t \ll MTF$. This can be achieved in the present situation by breaking the computer into many modules, each of which

* A nonredundant system is one which fails if any single element in the system fails. The exponential failure law for nonredundant systems has been justified for a wide class of complex systems for periods of observation which are short compared with the mean-time-to-failure of an individual component.[7]

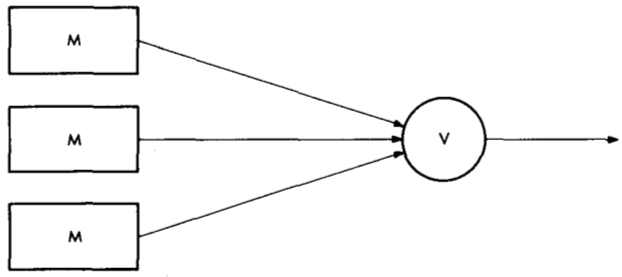



*Figure 1* **Triple redundancy as originally envisaged by Von Neumann.**

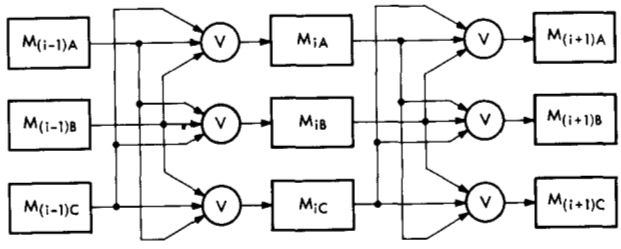



*Figure 2* **Triple-modular-redundant configuration.**

is much more reliable than the entire computer. In this instance $t \ll MTF$, and the triply redundant module is much more reliable than the original module. If these triply redundant modules are now reconnected to assemble an entire triple-modular-redundant (TMR) computer, an over-all improvement in the reliability of the computer will be achieved.

## Triple-modular redundancy with perfect voting circuits

Figure 2 illustrates the triple-modular-redundant configuration that will be used in this analysis. This configuration differs from the one shown in Fig. 1 because it employs three identical voting circuits instead of the one voting circuit previously used.[8] If it is assumed that the voting circuits do not fail, the two configurations have identical reliability. Later, when the unreliability of the voting circuits is taken into account, it will be observed that the voting circuits themselves are redundant in the configuration of Fig. 2. Hence single voting circuit failure will not necessarily cause computer failure.

The following assumptions are made:

1) The nonredundant computer is divided into $m$ modules.
2) Each module has just one input and one output.
3) The voting circuits do not fail.
4) The failures of the modules are statistically independent.
5) The modules $m$ are equally reliable.

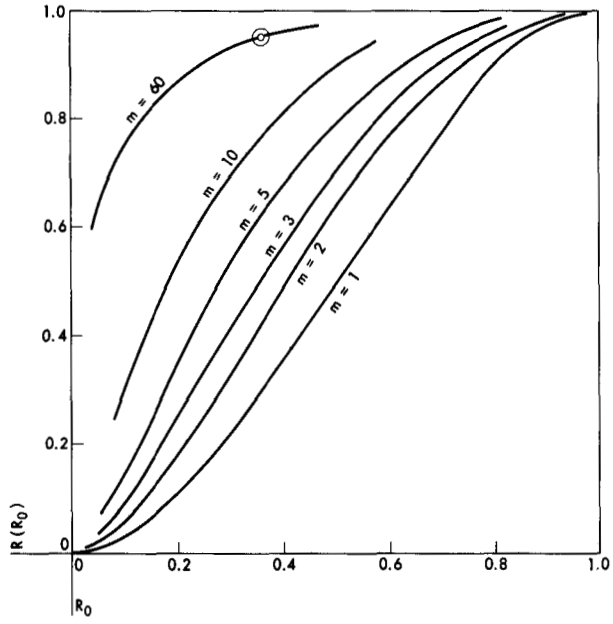



*Figure 3* **TMR reliability $R$ vs nonredundant reliability $R_0$.**
$R = (3R_0^{2/m} - 2R_0^{3/m})^m$.

Letting $R_0$ represent the reliability of the entire nonredundant computer and $R_M$ the reliability of a single module, assumption 5 implies

$$R_M = R_0^{1/m}. \tag{4}$$

Substituting this value for $R_M$ into Eq. (1) to find $R_T$, the reliability of one trio (a group of three modules connected as in Fig. 2), gives

$$R_T = 3R_0^{2/m} - 2R_0^{3/m}. \tag{5}$$

Reassembling modules into a complete computer having the same capability as the original nonredundant computer results in

$$R(R_0, m) = R_T^m = (3R_0^{2/m} - 2R_0^{3/m})^m, \tag{6}$$

where $R(R_0, m)$ is the reliability of the TMR computer.

Figure 3 shows TMR reliability $R$ plotted versus nonredundant reliability $R_0$ with $m$ as a parameter, Eq. (6). Note that as a consequence of assumption (3), the TMR reliability can be made as close to unity as one pleases by making an increasingly finer modular breakdown (a larger and larger $m$).

As an example, consider a nonredundant computer with an exponentially decaying reliability whose operating time is required to be equal to its *MTF*. Figure 3 (note circled point), illustrates that a TMR reliability of 0.95 can be achieved by breaking this computer into 60 modules.

### Analysis with imperfect voting circuits

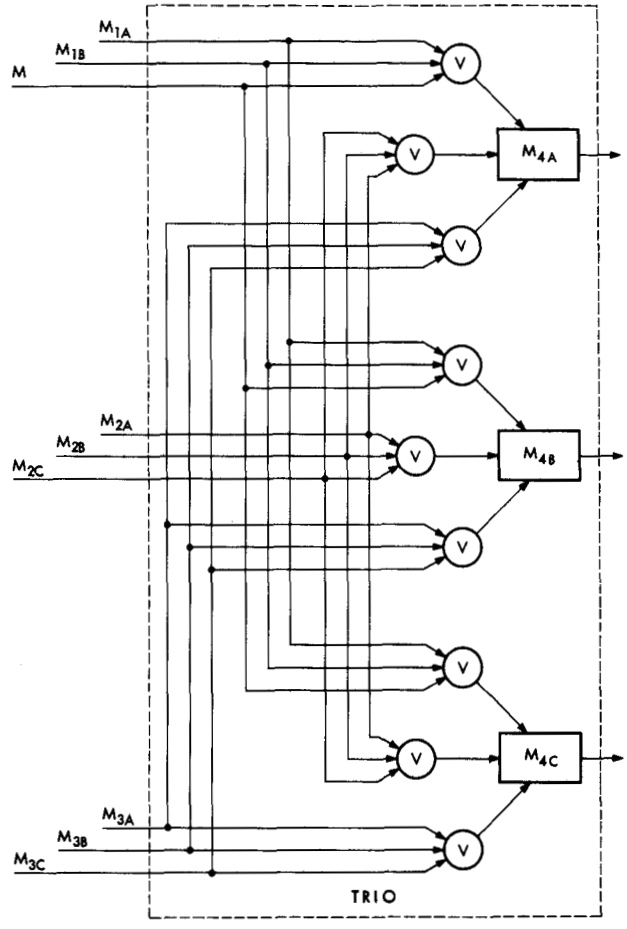
For the present, assumption (5) (the modules are equally reliable) will be retained. Assumption (2) (one input) can be relaxed by the following procedure.

Consider Fig. 4 which illustrates a section of a computer consisting of a trio having multiple inputs. It is convenient to group the voting circuits with the equipment driven, and to treat the reliability of the equipment within the dotted line of Fig. 4 as a unit. Having grouped the voting circuits with the following modules, the designation *trio* is retained for the equipment within the dotted lines and the relation $R = R_T^m$ still holds.

The assumption of only one output per module is not severe. In a real computer, nearly all modules do have only one output. For those modules which have more than one output, the conservative assumption is made that the failure of any one component of the module fails all outputs. The error introduced by this assumption is negligible.

To calculate the effect of unreliability of voting circuits, assumptions (1) and (5) of the preceding section are retained, but assumptions (2), (3) and (4) are replaced

*Figure 4* **Section of a computer, consisting of a trio of modules, which has multiple inputs.**



by the following assumptions:

2*) All module interconnections are made through input voting circuits as illustrated in Fig. 4.

3*) The failures of modules and voting circuits are statistically independent events.

4*) The voting circuitry associated with each module has the same reliability for all modules.

$R_V$ is defined as the reliability of the voting circuitry which drives a single module. For example, in Fig. 2 $R_V$ is the reliability of one voting circuit, while in Fig. 4 it is the reliability of three voting circuits. In practice $R_V$ will be close to unity, and in fact a reasonable value for $R_V$ is about 0.999 for operating periods on the order of one or two hundred hours using present-day circuits and components.

The effect of $R_V$ on the reliability of the trio can be taken into account by multiplying the reliability $R_M (= R_0^{1/m})$ by $R_V$. This is valid because by assumption (2*) the voting circuits are in reliability series with the following module, and the effect of voting circuit failure is the same as the effect of module failure. The result of substituting $R_V R_0^{1/m}$ in place of $R_0^{1/m}$ in Eq. (6) is

$$R(R_0, R_V, m) = (3R_V^2 R_0^{2/m} - 2R_V^3 R_0^{3/m})^m . \qquad (7)$$

It is of interest to consider the reliability of the TMR computer as the number of modules $m$ becomes large. Figure 3 indicates that $R$ increases monotonically with increasing $m$, but this figure is for the case of $R_V = 1.0$. If the voting circuits are not perfectly reliable, it is logical to expect that as $m$ is made very large, thereby requiring many voting circuits, the TMR computer reliability will eventually begin to decrease. In fact, it can be shown that the limit of $R(R_0, R_V, m)$ as $m$ approaches infinity is zero.

For small $m$, the unreliability of the voting logic will have little effect on the reliability of the TMR computer because a small number of voting circuits are needed and because $R_V$ is close to unity.

The curves of Fig. 3 can be expected to be valid then for small $m$. Since $R$, considered as a function of $m$, initially increases with increasing $m$, but ultimately approaches zero as $m$ approaches infinity, the function has a maximum for some value of $m$, say $m_0$. Figure 5, which is a plot of Eq. (7) when $R_0 = 0.37$ and $R_V = 0.999$, is an example of the shape of the graph of the function $R(m)$.

To find an approximation for $m_0$, which is that value of $m$ which maximizes $R(R_0, R_V, m)$, in terms of $R_0$ and $R_V$, we use the fact that $R_V$ is near unity in the practical case and that $R_M = R_0^{1/m}$, the reliability of a single module, must be near unity if a high TMR reliability is to be achieved. As a consequence the quantity

$$X = \ln(R_V R_0^{1/m}) \qquad (8)$$

will be small in practice. On the other hand, it follows from Eq. (7) that $R$ and hence ln $R$, is a function of $X$.

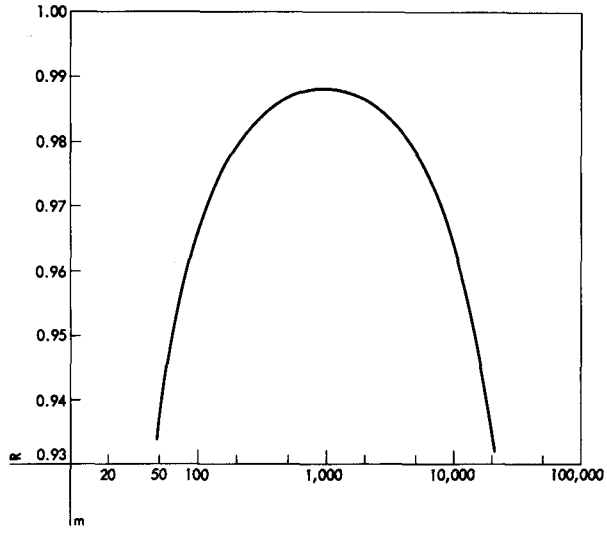



*Figure 5* **Example of the function R(m).** $R_0 = 0.37$; $R_v = 0.999$; *and* $R(m) = [3R_0^{2/m} R_v^2 - 2R_0^{3/m} R_v^3]^m$.

Thus an approximation to $R$ can be obtained by expanding ln $R$ into a power series in $X$ and retaining the first non-zero term only. This leads to the following approximation for $R$:

$$R \approx \exp(-12 \ln R_V \ln R_0) \times \exp\{-3[\sqrt{m} \ln R_V - (\ln R_0/\sqrt{m})]^2\} . \qquad (9)$$

Noting that the first factor in (9) is independent of $m$, and that the maximum value of the second factor is $e^0 = 1$, the conclusions are reached that

$$R_{max} = \exp[-12 \ln R_V \ln R_0] , \qquad (10)^*$$

and that the value of $m_0$ is given by

$$m_0 = \ln R_0 / \ln R_V . \qquad (11)^*$$

As an example, consider the case of a nonredundant computer and voting circuits with exponentially decaying reliabilities. Then

$$\ln R_0 = -f_0 t , \qquad \ln R_V = f_v t , \qquad (12)$$

where $t$ is the operating time and where $f_0$ and $f_v$ are the (constant) failure rates of the nonredundant computer and the voting circuitry associated with each module respectively.

Now (11) can be written as

$$m_0 = f_0 / f_v . \qquad (13)$$

Hence for the case of constant failure rates, the

* Equations (10) and (11) can be written to a higher degree of approximation as follows:

$$R_{max} = \exp\langle -12 \ln R_V R_0 \{[1 - (2/3)\ln R_V]/[1 - 2 \ln R_V]^2\}\rangle \qquad (10a)$$

$$m_0 = \{[\ln R_0][1 + \ln R_V]\}/\{[\ln R_V][1 - (7/3)\ln R_V]\} \qquad (11a)$$

optimum value of $m$ is, to a first approximation, independent of the equipment operating time $t$.

Equation (13) can be rewritten

$$f_v = f_0/m_0 , \tag{14}$$

which states that for maximum reliability each module must have approximately the same failure rate as the voting circuitry which drives it. Under fairly general assumptions, the failure rate of nonredundant equipment is proportional to the size or complexity or component count of the equipment. Thus for maximum reliability, the modules into which the computer is subdivided must have the same size as the voting circuitry which drives them. Hence, it follows that for maximum reliability the total size of all modules should be equal to the size of all voting circuitry. However, by definition, the size of all modules is three times the size of the original nonredundant computer. Therefore, for maximum reliability, the size of the TMR computer must be approximately six times the size of the original nonredundant computer.

For practical reasons, the advisability of using a procedure which requires six times as much equipment may be questionable. The factor of six is based on the use of the optimum $m$. However, even for a nonredundant reliability which is an arbitrary function of operating time, near optimum TMR reliability can be achieved by utilizing a value of $m$ considerably smaller than the optimum value $m_0$.

Substituting (10) and (11) into (9) yields the following formula:

$$\ln R = \left[1 + \frac{1}{4}\frac{(m - m_0)^2}{mm_0}\right] \ln(R_{\max}) . \tag{15}$$

Since the natural logarithm of the reciprocal of a reliability which is close to unity is approximately equal to the probability of failure, it is seen from (15) that the probability of failure for a TMR computer is approximately

$$1 + \frac{1}{4}\frac{(m - m_0)^2}{mm_0}$$

times the minimum attainable failure probability. For instance, when $m = (1/2)m_0$, the failure probability is 1.125 times the minimum failure probability. The same result is obtained when $m = 2m_0$.

For the above example, (15) provides a trade-off curve between size and reliability of the TMR computer; namely, if the size of the redundant computer is $3(1 + K)$ times the size of the nonredundant computer, then the failure probability of the redundant computer is

$$1 + \frac{1}{4}\frac{(1 - K)^2}{K}$$

times the minimum failure probability.

As a numerical illustration, the above equations are now applied to the problem of optimizing the TMR reliability of a nonredundant computer with exponentially decaying reliability whose operating time is required to be as great as its mean-time-to-failure. Then $R_0 = e^{-1} = 0.368$. Using an estimated value of 0.999 for $R_V$ in (10) and (11), one finds that the optimum number of modules is $m_0 = 1000$, and the corresponding maximum TMR reliability is $R_{\max} = 0.988$. To achieve $R = 0.95$, one can show from (15) or from Fig. 5 that $m = 0.0674m_0 = 67$ modules. (Note that this is only slightly higher than the 60 modules previously calculated on the basis of $R_V = 1.0$.) The size of the TMR computer in this case is $3 + 3(.0674) = 3.20$ times the size of the nonredundant computer.

## Modules with unequal reliabilities

In this section the assumptions 1, 2*, and 3* made previously are retained, but the unnecessarily restrictive assumptions 4* and 5 will be dropped.

The reliability, $R_K$, of the $K$-th module can be presented by $R_K = R_0^{1/m} \exp \Delta_K$, where $\Delta_K$ measures the deviation of $R_K$ from the reliability $R_0^{1/m}$, which would have prevailed for modules of equal reliability. Likewise, the reliability of the voting circuitry associated with the $K$-th module can be written as $R_V \exp \delta_K$, where $R_V$ is the average reliability of the voting circuitry associated with the modules. Thus the sum of the $\Delta_K$'s and the sum of the $\delta_K$'s are zero.

It can be shown that formula (9) must now be replaced by

$$R = \exp\langle -3m\{[(\ln R_0)/m] + \ln R_V\}^2\rangle \times \exp\left[-3\sum_K (\Delta_K + \delta_K)^2\right] . \tag{16}$$

Hence, unequal reliabilities of modules and associated voting circuitry will lower the attainable TMR reliability. In many cases of practical interest the total amount of voting circuitry will be a small portion of the entire computer equipment. In that event $\delta_K$ can be expected to be small compared to $\Delta_K$, and, hence can be neglected in (16). Under these circumstances it follows that for maximum TMR reliability, the nonredundant computer should be subdivided into modules of as nearly equal reliability as possible.

## The problem of module interconnection

In the preceding section it has been assumed that every input to every module is checked by voting circuitry (assumption 2*). One may question whether this procedure will indeed lead to maximal TMR reliability for a realistic situation.

The basic problem is that in a practical computer the output of some highly reliable module, such as a timing pulse generator, can be an input to a large number, say $n$, of the other modules. Rigorous application of the principles followed in the preceding sections would require not only triplication of the

timing pulse generator, but would also demand the installation of $3n$ voting circuits to check the output of the three timing pulse generators. It would seem possible that for large $n$ this procedure could decrease rather than increase the TMR reliability.

To analyze this problem the rigorous application of the TMR principle as in Fig. 4 and three alternative configurations are compared. Conditions for the values of certain critical reliabilities are derived which determine whether over-all reliability is enhanced or degraded by replicating a given unit. The interpretation of the conditions, which are stated in the form of inequalities, should take into account that reliability is often not known very precisely. Thus, while the mathematical treatment yields well-defined regions where replication should or should not be used, in practice there is usually a rather broad range over which nearly equal over-all reliability will result in either case.

The four cases to be compared are:

Case 1: Connection of $n$ triplicated voting circuits at the inputs to the $n$ modules driven by the timing pulse generator.

Case 2: Connection of one triplicated voting circuit at the output of the timing pulse generator.

Case 3: Omission of the voting circuits, retaining the triplicated timing pulse generator.

Case 4: Use of a single timing pulse generator to drive all $n$ triplicated modules without voting circuitry.

For convenience in further discussion, the following terms are defined:

*M-unit*: An $M$-unit is a trio of modules whose outputs are connected to other trios only through voting circuits.

*Q-unit*: A $Q$-unit is a trio of modules whose outputs are connected to other trios directly.

*N-unit*: An $N$-unit is a non-triplicated module.

Use of this terminology and the definitions of the four cases are illustrated in Fig. 6. The following formulas for the reliabilities of the four cases can be derived:

$$R_1 = (3R_{tpg}^2 - 2R_{tpg}^3) \prod_{i=1}^{n} (3R_i^2 R_V^2 - 2R_i^3 R_V^3) \quad (17)$$

$$R_2 = [3R_{tpg}^2 - 2R_{tpg}^3] \times \left[ R_V^3 \prod_{i=1}^{n} (3R_i^2 - 2R_i^3) + 3R_V^2(1 - R_V)R^2 \right] \quad (18)$$

$$R_3 = R_{tpg}^3 \prod_{i=1}^{n} (3R_i^2 - 2R_i^3) + 3R_{tpg}^2(1 - R_{tpg})R^2 \quad (19)$$

$$R_4 = R_{tpg} \prod_{i=1}^{n} (3R_i^2 - 2R_i^3) , \quad (20)$$

where $R_{tpg}$ = reliability of one tpg module
$R_V$ = reliability of one voting circuit
$R_i$ = reliability of the $i$th driven module $(i = 1, 2, \cdots, n)$
$R = \prod_{i=1}^{n} R_i$ = aggregate nonredundant reliability of modules driven.

In the analyses which follow it is assumed that the failure probabilities of the timing pulse generator and of the voting circuits are sufficiently small that second and higher order terms can be neglected compared with first-order terms. It is also assumed that the failure probability of the aggregate of all modules driven is much larger than timing pulse generator and voting circuit failure probability.

### *Comparison of $R_1$ and $R_2$*

One can show from (17) and (18) that $R_1 > R_2$ if and only if

$$1 - R_{V1} < [(1 - R^2)S + 2 \ln R]/n , \quad (21)$$

where $R_{V1}$ is the reliability of a voting circuit for case 1, $R_{V2}$ is the reliability of a voting circuit for case 2, and $S = (1 - R_{V2})/(1 - R_{V1})$ is the ratio of voting circuit failure probabilities.

If the same standard voting circuit is used in both case 1 and case 2, then $S = 1$. However, normally $S > 1$ because output power requirements usually demand more voting circuitry to drive $n$ trios than to drive one trio.

It is easy to show that for $S = 1$, (21) implies $R_2 > R_1$, in all cases. It is of interest to solve inequality (21) for $S$ to determine that value of $S$ for which $R_1$ and $R_2$ are equal. As an illustrative example, consider case 1 when the reliabilities of the driven modules are equal to each other and are also equal to the voting circuit reliabilities—an example previously shown to yield optimum TMR reliability. In this case, $R_1 > R_2$ if and only if

$$S > \frac{3 \ln(1/R)}{1 - R^2} .$$

As a continuation of a previous numerical example, if $R$, the nonredundant reliability of all modules driven is $1/e$, then $R_1 > R_2$ if and only if $S > 3.47$. Hence, in this specific example, the configuration of case 1 is more reliable than that of case 2 if and only if the voting circuitry required to drive $n$ modules is more than about $3\frac{1}{2}$ times as likely to fail as the voting circuitry required to drive one module.

### *Comparison of $R_2$ and $R_3$*

Since $(3R_{tpg}^2 - 2R_{tpg}^3)$ is extremely close to unity in (18), inspection of (18) and (19) reveals that $R_3 > R_2$ if and only if

$$R_{tpg3} > R_{V2} . \quad (22)$$

Thus, the configuration of case 3 is more reliable

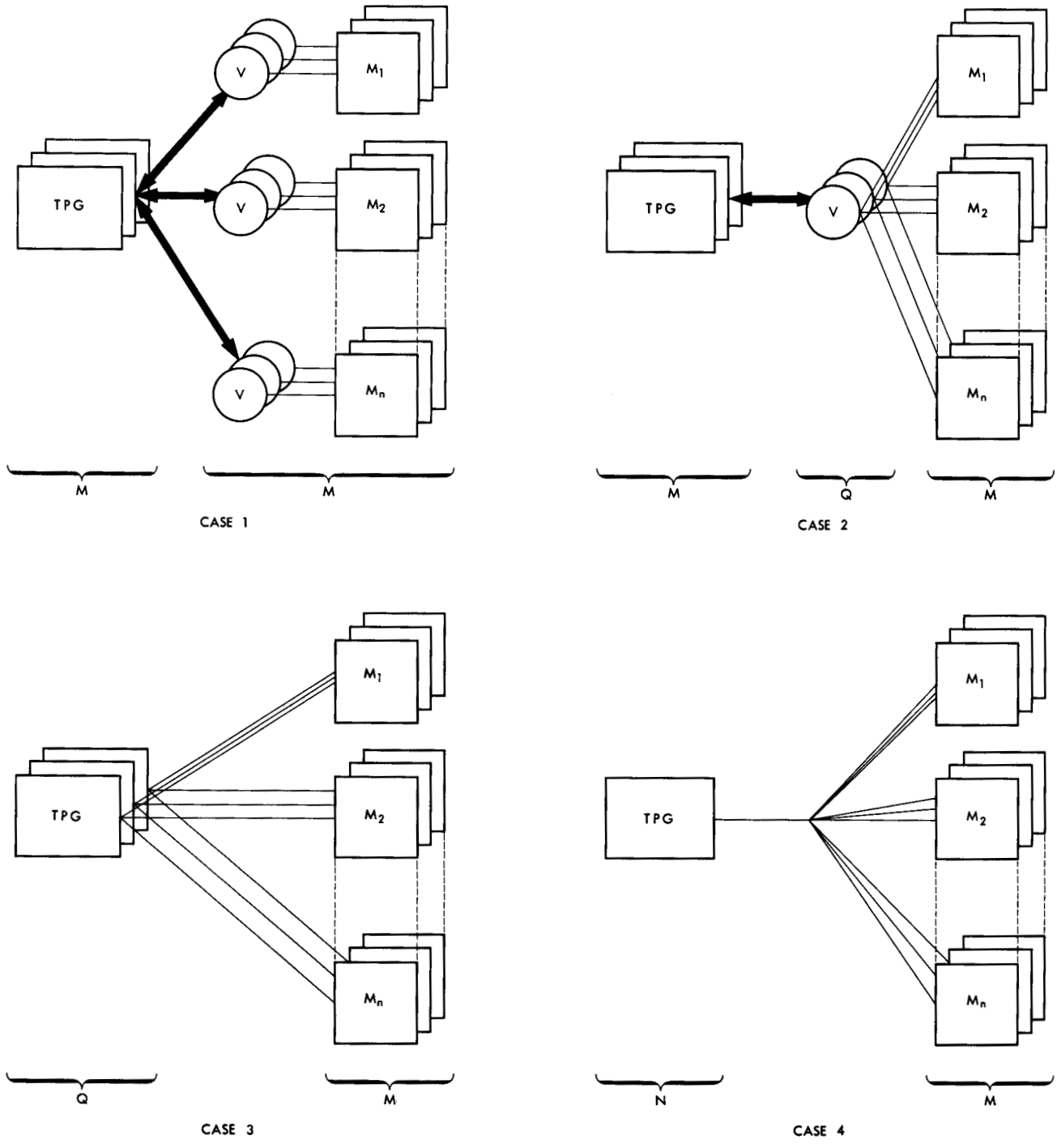



*Figure 6* **Solutions to problem of unbalance between amount of computer logic and voting logic.** ↔ *implies that modules are connected to voting circuits as in Fig. 2.*

than that of case 2 if and only if the reliability of the timing pulse generator of case 3 exceeds the reliability of the voting circuit of case 2.

### *Comparison of $R_3$ and $R_4$*

One can show from Eqs. (19) and (20) that $R_4 > R_3$ if and only if

$$1 - R_{tpg4} < 3(1 - R_{tpg3})(1 - R^2)\,, \tag{23}$$

where $R_{tpg3}$ and $R_{tpg4}$ are the timing pulse generator reliabilities for the two cases.

It is instructive to compare $R_4$ and $R_3$ for the special

case that the exponential law of reliability holds, i.e.,

$$R = e^{-t/MTF}$$

$$R_{tpg3} = e^{-f_3 t}$$

$$R_{tpg4} = e^{-f_4 t} .$$

In this case one finds that $R_4 > R_3$ if and only if

$$t > \frac{1}{2} \ln \frac{3f_3}{3f_3 - f_4} (MTF) . \tag{24}$$

Further specializing to the case that $f_3 = f_4$, one finds $R_4 > R_3$ if and only if

$$t > 0.202\ (MTF) . \tag{25}$$

Thus for short operating times the configuration of case 3 is more reliable, and for long operating times that of case 4 is more reliable. Inequality (25) gives the *cross-over* time at which the two configurations are equally reliable.

### Monte Carlo analysis of a TMR computer

The preceding results provide some guide lines for the design of a TMR computer. Although these guide lines are useful, they exhibit several deficiencies. First, they are unable to cope in detail with the complicated computer designs met in practice. Secondly, they are qualitative only, since they indicate rules for improving reliability without furnishing a quantitative measure for this improvement. Clearly, additional tools are desirable in the design of a TMR computer. One such tool and its usage will be described in this section. It consists of a Monte Carlo model for simulating the statistical failure structure of a TMR computer.

The model requires a list of all modules $m_i$; $i = 1, \cdots, s$ which make up the TMR computer, together with a list, indicating the triplets among the modules which form $M$-units, the triplets which form $Q$-units, and those single modules which are $N$-units. The simulation of the logical structure of the TMR computer is completed by listing for each module the set of those modules which feed it.

This part of the model enables us to decide whether the TMR computer fails when the set of failed modules is given. For example, if a failed module is an $N$-unit, the TMR computer has failed. However, if a failed module belongs to an $M$-unit, whose other two modules are still operating, then this failed module will not affect the TMR computer operation so long as the three modules feed into operating voting circuits. Thus it is possible to supplement the model by a simple decision process which, for a given sequence of failed modules, determines the first point in the sequence where failure of the TMR computer occurs.

The model can now be used to play the following "game". At the beginning of the game consider all modules to be operating. Select modules successively and declare them permanently failed, considering those not yet selected as still operating. At each stage in this succession decide whether the TMR computer fails as a result of the modules declared failed up to that stage. The game ends when failure of the TMR computer occurs for the first time. The result of a single game is the number of stages in the sequence, in other words, the number of modules necessary to cause computer failure in this game.

By properly selecting the sequence of failed modules and by playing a large number of games, one can simulate the failure structure of the TMR computer. To obtain a selection process every module should be looked upon as a collection of components (transistors, diodes, capacitors, resistors, soldered or welded connections). Failure of a component shall imply failure of the module of which it is a part. Given a configuration of failed and operating components, and given the occurrence of one additional component failure, the probability that this component belongs to module $m_i$ shall be denoted by $p_i$. Thus $\sum_{i=1}^{s} p_i = 1$. Although, in general, the probability $p_i$ will be a function of operating time and of the configuration of failed and operating components just prior to the additional component failure, it shall be assumed, for the sake of simplicity, that the probabilities $p_i$ are constant. Thus, for each game, the sequence of failed modules can be selected as follows.

Starting with all components operating, each stage of the sequence represents an additional component failure. By casting a die with $s$ faces, $m_1, m_2, \cdots, m_s$, such that face $m_i$ has probability $p_i$ to occur, the module in which that component failure occurred is determined. Note that repetitions in the sequence, though unlikely, are possible. Such repetitions indicate modules in which more than one component failure has occurred. Hence, the result of a game is now the number of component failures necessary to fail the TMR computer in this game. By playing a large number of games and by tabulating the percentage of the games which terminated after exactly $k$ component failures, one obtains an estimate for the probability, $P_k$, of a TMR computer failure after exactly $k$ component failures.

To obtain a model in which the probabilities $p_i$ are approximately constant, assume that all components have exponentially decaying reliabilities. It can be shown that for computer designs met in practice, the above assumption implies that $p_i$ is approximately constant and can be estimated by dividing the sum of the failure rates of the components in module $m_i$ by the sum of the failure rates of all components.

The Monte Carlo model shall now be completed with a model for computing the TMR reliability as a function of operating time $t$. Assuming a Poisson distribution for the number, $k$, of component failures after exactly $t$ hours of operation, it is seen that the probability of $k$ or more component failures in $t$ hours of operation is given by

$$e^{-m} \sum_{\gamma=k}^{\infty} m^{\gamma}/\gamma! , \tag{26}$$

where $m$ is the product of $t$ and the sum of the failure rates of all components. Multiplication of (26) by $P_k$ yields the probability of computer failure before $t$ hours of operation as a consequence of exactly $k$ component failures. Hence, the TMR computer reliability, $R(t)$, i.e., the probability of TMR computer operation for at least $t$ hours, is given by

$$R(t) = \sum_{k=1}^{\infty} P_k e^{-m} \sum_{\gamma=0}^{k-1} m^{\gamma}/\gamma! \,. \tag{27}$$

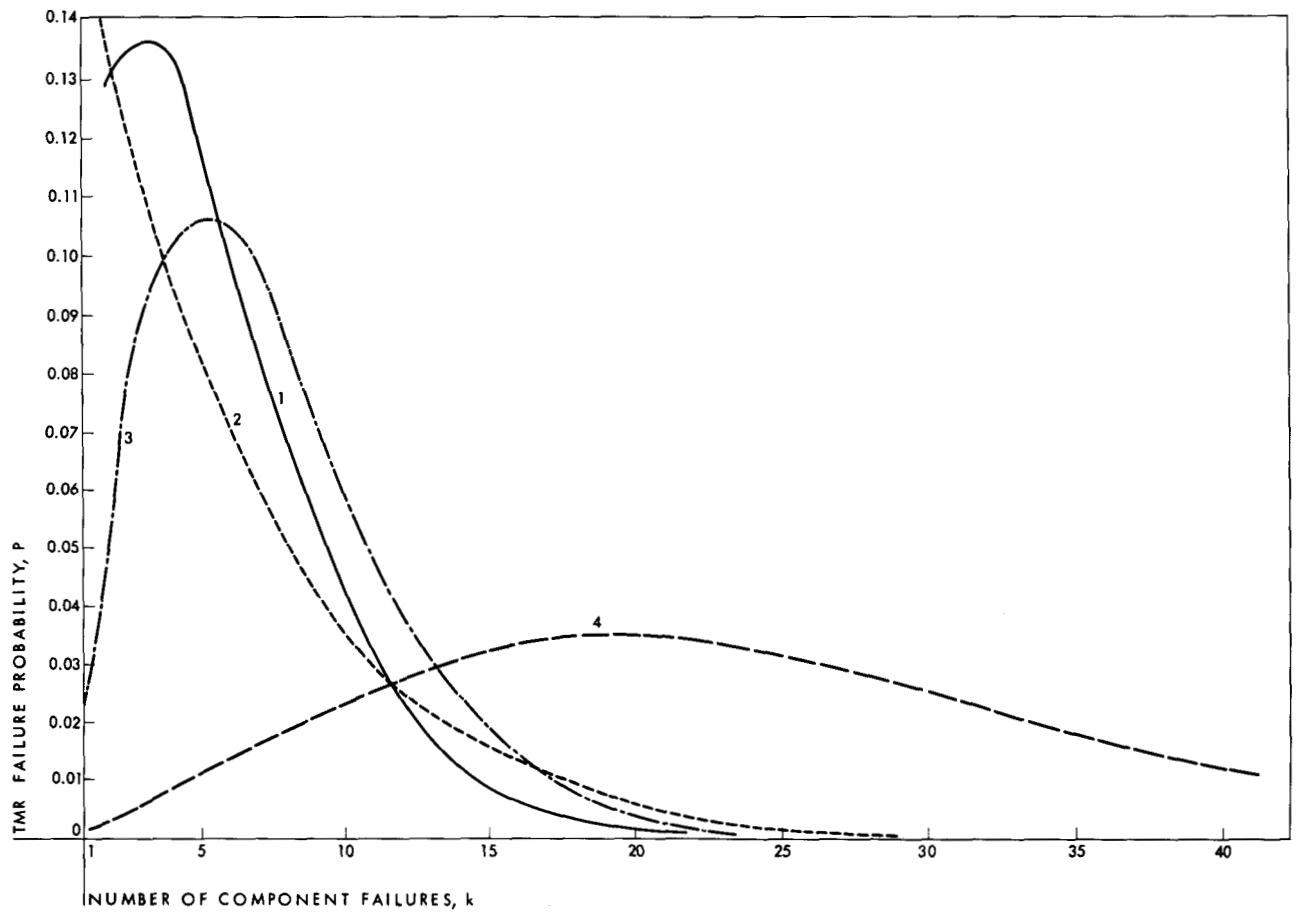
Note that the above Monte Carlo model is capable of furnishing the TMR computer failure probability as a function of the number of component failures and of furnishing the TMR computer reliability as a function of the operating time $t$. As a numerical example, TMR failure probability versus the number of component failures, and the TMR reliability as a function of $t$ have been displayed in Figs. 7 and 8 for four different TMR computer designs. All four TMR designs were based on the same nonredundant computer whose reliability has been depicted in Fig. 8.

The curves labeled 1 in Figs. 7 and 8 correspond to a reasonably realistic TMR implementation using $Q$- and $M$-units only. The curves labeled 2 correspond to the TMR implementation obtained by retaining the $M$-units and the corresponding voting circuitry, but by replacing all $Q$-units by $N$-units. A deterioration of the reliability of design 2 as compared to design 1 is clearly evident in Fig. 7 as well as Fig. 8. Design 3 is based on the results obtained in the preceding section (Eq. 24) which were used to decide which $Q$-units in design 1 should remain $Q$-units and which $Q$-units should be converted to $N$-units. The resulting reliability improvement of design 3 over designs 1 and 2 is clearly noticeable in Figs. 7 and 8. Design 4 was a purely TMR design, consisting of $n$ $M$-units, $n$ being equal to the total number of $Q$- and $M$-units in design 1. All $M$-units were chosen equal, in such a manner that the nonredundant computer corresponding to design 4 possessed the same reliability as the nonredundant computer on which designs 1, 2 and 3 were used. It must be emphasized that design 4 totally disregards the logical design of the nonredundant computer of the designs 1, 2, and 3. The reason for including design 4 is to demonstrate the power of a pure TMR design. Unfortunately such a design seems rarely attainable in view of the complex logical constraints which must be met in practice.

## Conclusion

Since maintenance during computer operating time will not be permitted in many applications of interest,

*Figure 7* **Monte Carlo results, showing TMR failure probability vs number of component failures.**



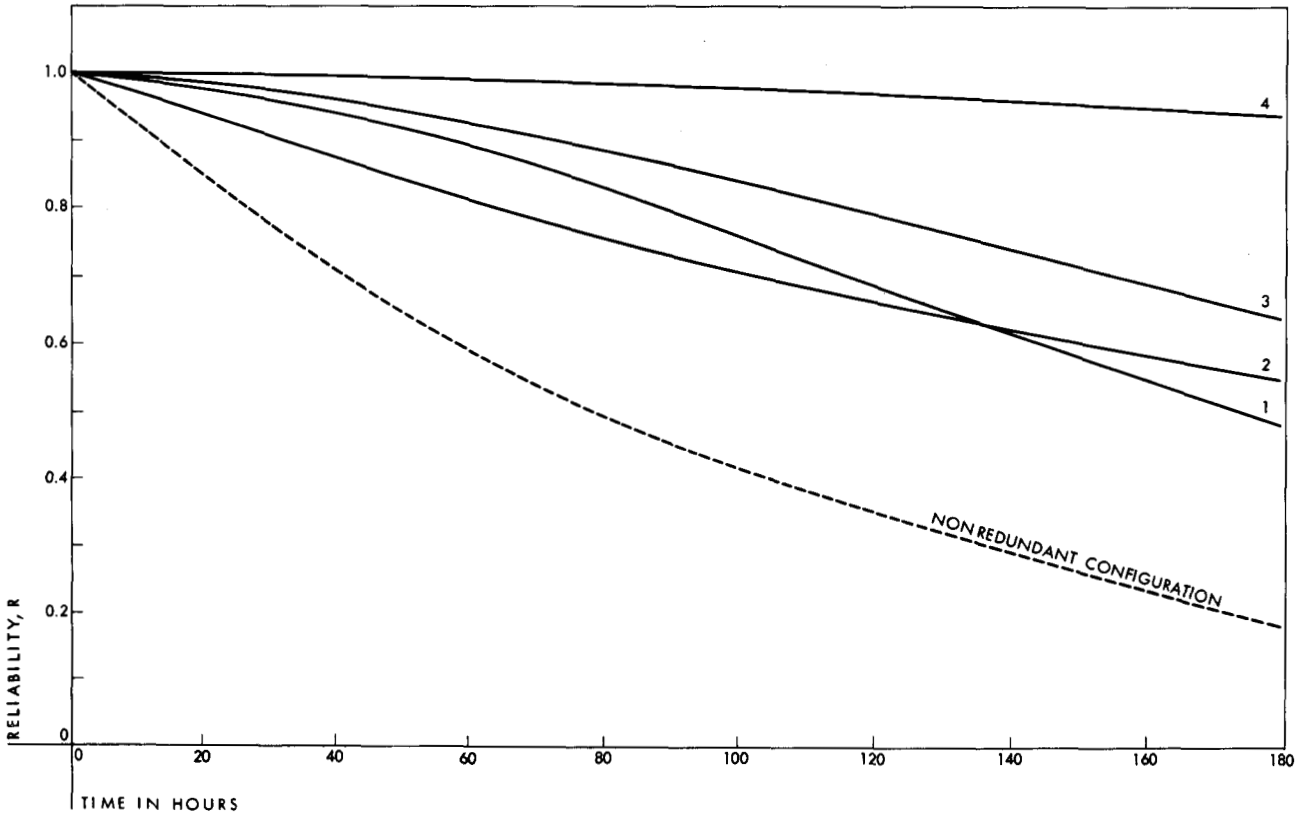



*Figure 8* **Monte Carlo results, showing reliability vs time.**

a reliability which would make in-service maintenance unnecessary has been the goal. It is worth noting, however, that the TMR concept lends itself very well to a maintenance situation. Intermittent component failures will not normally cause the TMR computer to fail. By a suitable modification of the voting circuits, it is possible to detect and indicate where intermittent as well as permanent component failures have occurred. It is even possible to replace faulty units while the TMR computer continues to operate without error.

The preceding analysis has concerned itself with the logical structure of a digital computer. However, the TMR concept is applicable to any digital system, mechanical or electrical. In particular it can be applied to storage media and input-output equipment associated with digital systems.

## Acknowledgment

The authors are indebted to G. Cour, who provided the hypothetical TMR computer and who conceived and performed the preliminary Monte Carlo analysis; they also thank R. Eckstrom, who provided the 704 program for the Monte Carlo analysis.

## References

1. J. Von Neumann, "Probabilistic Logics," *Automata Studies*, Princeton University Press, 1956.
2. H. H. Goldstine, "Some Remarks on Logical Design and Program Checks," *Proc. EJCC*, Washington, D.C., Dec., 1953, pp. 96-98.
3. E. F. Moore and C. Shannon, "Reliable Circuits Using Less Reliable Relays," *J. Franklin Inst.*, **262**, 191-208, 281-297 (1956).
4. W. E. Dickinson and R. M. Walker, "Reliability Improvement by the Use of Multiple-Element Switching Circuits," *IBM Journal* **2**, No. 2, 142 (1958).
5. B. J. Flehinger, "Reliability Improvement Through Redundancy at Various System Levels," *IBM Journal* **2**, No. 2, 148 (1958).
6. D. E. Rosenheim and R. B. Ash, "Increasing Reliability by the Use of Redundant Machines," *IRE Trans. on Elec. Comp.*, **EC-8**, No. 2, 125 (1959).
7. D. R. Cox and W. L. Smith, "On the Superposition of Renewal Processes," *Biometrika* **4**, No. 1, 99 (1954).
8. M. Cohn, "Redundancy in Complex Computers," *Proceedings of the National Conference on Aeronautical Electronics*, Dayton, Ohio, May, 1956, pp. 231-235.

---

*Note added in proof:* Since submission of the original manuscript, the following additional pertinent papers have appeared in the literature:

1. W. G. Brown, J. Tierney and R. Wasserman, "Improvement of Electronic-Computer Reliability Through the Use of Redundancy," *IRE Trans. on Elec. Comp.*, EC-10, No. 3, 407 (1961).
2. G. Buzzell, W. Nutting and R. Wasserman, "Majority Gate Logic Improves Digital Systems Reliability," *1961 IRE National Convention Record, Part II*, p. 264.
3. W. C. Mann, "Systematically Introduced Redundancy in Logical Systems," *1961 IRE National Convention Record, Part II*, p. 241.
4. S. Muroga, "Preliminary Study of the Probabilistic Behavior of a Digital Network with Majority Decision Elements," Rome Air Development Center, RADC-TN-60-146, August 1960.
5. L. A. M. Verbeck, "Reliable Computation with Unreliable Circuitry," *Proceedings of the First Bionics Symposium*, Sept. 1960, pp. 83-92.

*Original manuscript received September, 1959*
*Revised manuscript received August 15, 1961*