



Modular computer architecture strategy for long term missions

by F. D. ERWIN

Hughes Aircraft Company
Fullerton, California

and

E. BERSOFF

NASA Electronics Research Center
Cambridge, Massachusetts

INTRODUCTION

Long term mission reliability of a modular computer has been studied at Hughes Aircraft Company as a consequence of a study with NASA ERC.^{1,2} Particular interest lay in the attainment of long term reliability with modular computer organization and the effects on reliability of variations in modular organization. The results of this investigation are presented in this paper.

In the past, the designers of aerospace computers have concentrated on increasing computational speed and arithmetic capability within stringent weight and power limitations. There seems to be little doubt that aerospace computers will soon be extremely fast, versatile and compact. A requirement for long term system reliability has been developing and may drastically change the nature of the on-board computer. Extremely long missions are being planned which require a computer to operate for one to five or more years after launch. Current on-board computer systems are not adequate for this task.

One promising approach for achieving reliability and flexibility is through modular design, where independent physical modules, functionally organized (e.g. memory, arithmetic, control, Input/Output) can be added or deleted to adapt to the required performance

and processing needs in terms of speed and reliability. Improvement in reliability through the use of additional hardware has been receiving growing attention in the aerospace computer community.³

Specifically in this paper, a technique will be described which when properly applied will determine a computer configuration which can satisfy a required probability of mission success for some stated mission duration. It is assumed that some basic computer system exists which can perform the required computations; what remains is to determine which additional computers or sub-computers should be added to provide the necessary system reliability.

A modular computer design

Several techniques exist which are designed to increase the reliability of any given computer system. The approach taken here is to have a single computer perform all mission computations while a number of other computers remain in a dormant mode until the working computer fails. At that time, the failed computer is turned off and one of the dormant computers is turned on to resume the computations. The size, weight, and power restrictions will typically limit the number of spare computers that are available. An additional refinement to this concept is to segment

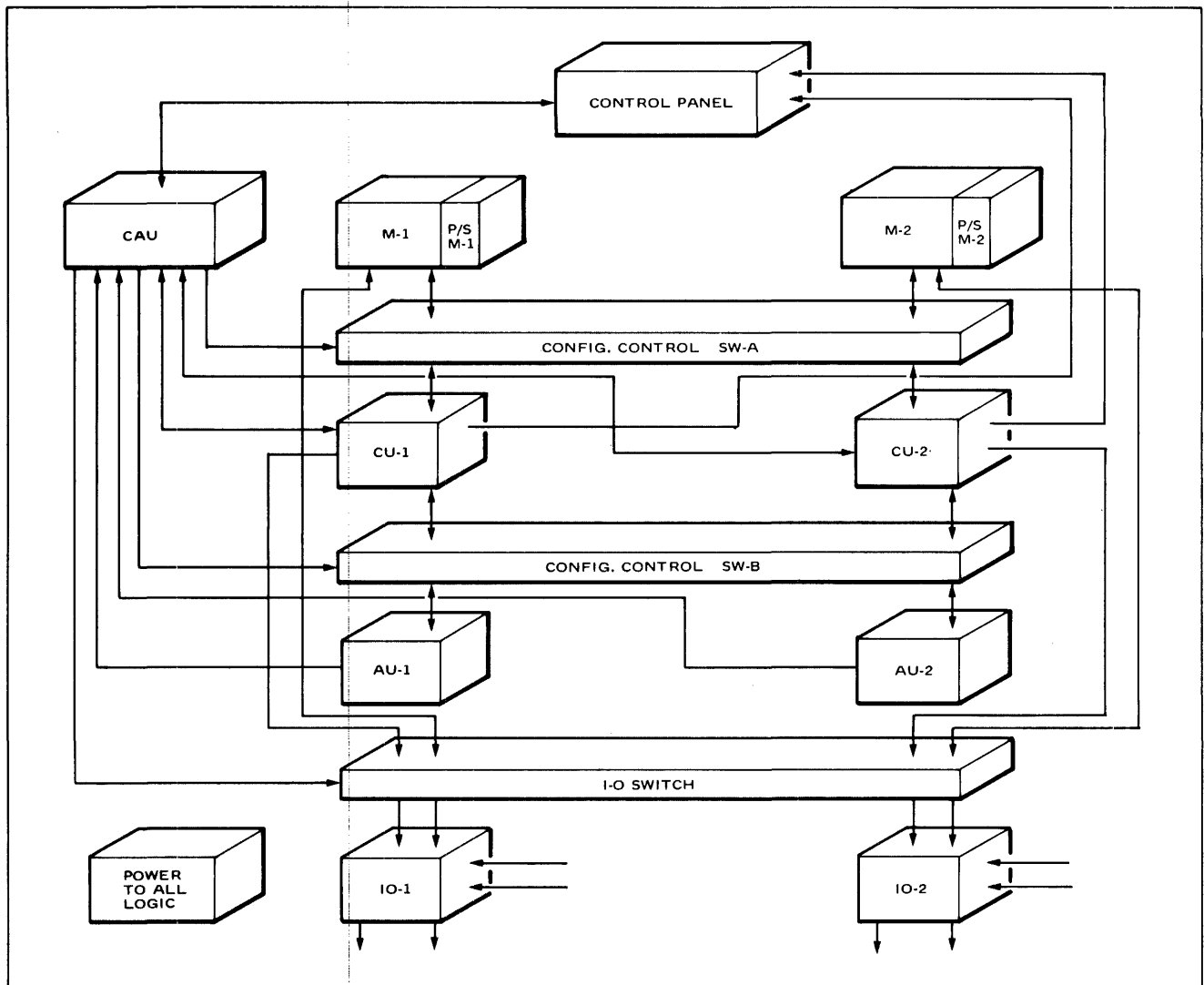


Figure 1—Modular computer breadboard

each computer into functional modules. One possible method is to isolate the memory, central processor, and I/O functions as in the Hughes H4400 computer. Another is to partition the computer into discrete memory, control, arithmetic, and I/O units, as in the NASA modular computer. This second approach will increase the total system parts count, but the increased modularity may ultimately enhance system reliability.

A mathematical model was constructed which permits the evaluation of computer reliability for the various configurations.

The two configurations mentioned above were normalized in terms of logic complexity to the NASA modular computer and the mission reliability was evaluated. Analysis showed that short term reliability

using voting techniques as would be done during boost phase, favors the H4400 modularity concepts whereas the long term reliability tends to favor the NASA modular computer partitioning. In either case, the long term reliability is very sensitive to the logic distribution within the modules and to the basic reliability of the components.

A breadboard modular computer of the MCB with two modules of each type (a two column system) is being constructed. These modules are of sufficient complexity to prove many of the points under consideration. First, the system may be configured in a one active, one standby fashion so that the techniques of error detection and reconfiguration may be explored. Figure 1 is a block diagram of the NASA computer

breadboard. The CAU (Configuration Assignment Unit) is the module that provides for continuation of system functioning under module or switch failure. It controls the activation and connection of standby modules into the operating system and failed modules out of it. The system can be configured so that one active string (memory, control, arithmetic, I/O) performs computations while the other unused modules remain in a standby state. If a failure occurs in any module, the module can be turned off and a standby module switched in. With the breadboard, if more than one of any distinct module fails, the system fails.

In order to compute the reliability of the modular computer, it is necessary to know the number of components in each module and their failure rates. Since it is also interesting to examine the case of the three modules (memory, central processor, I/O) configuration, Table I presents a breakdown of components for each case. As is shown in the figure, the basic component can be either the gate or the integrated circuit.

Enough experience with the NASA breadboard has been accumulated to instill a high level of confidence that the modular computer concept is sound and indeed workable. The system may be arranged in a TMR fashion during the boost phase of a mission when calculations are proceeding too rapidly to allow reconfiguration. After boost, two of the three computers would be turned off and the system would enter the one-active two standby mode. A block diagram of the proposed system appears in Figure 2. With the exception of the CAU which would be more complex due to the additional modules it must service, the component counts for each of other modules should be approximately the same as listed in Table I for the two column system. It will be shown that under some conditions even dual redundancy for the working modules would not provide adequate reliability for a five year mission. Several things can be done to eliminate this deficiency. First, and most fundamentally, the components (gates, IC's, or LSI's), can be made more reliable. Second, different configurations can be structured such as 4 column and others so as to enhance the mission reliability. Certainly, the nature of the mission and weight constraints will impose a limit on hardware launched. The control unit (or alternatively the central processor) is clearly the largest single module of the system. If only an additional control unit were carried, total reliability would be increased. It will be shown, however, for a five year mission, considerable reliability enhancement must be made to the CAU as well as improvements to the other modules.

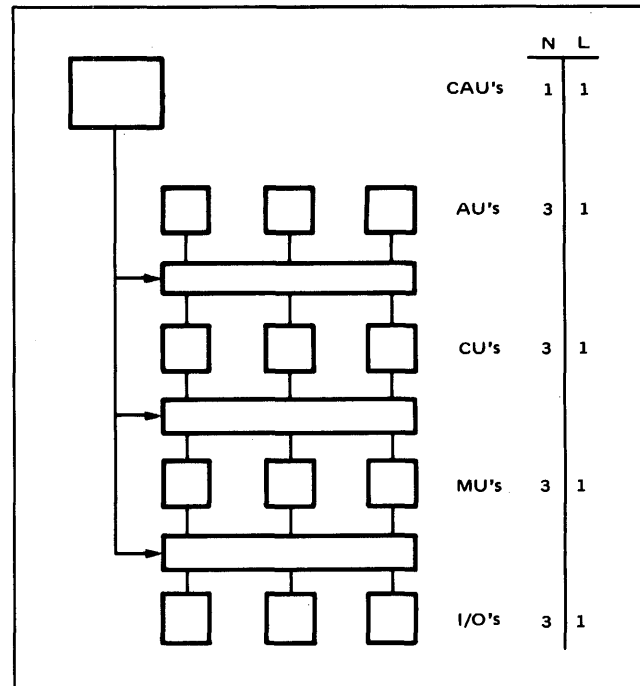


Figure 2—Three column modular computer

Several techniques designed to improve the CAU are required if the computer is to function for several years. One is to build the CAU with as few components as is possible; assuming equal component reliability the fewer components it has, the longer it will last. A further increase in reliability can be obtained by tripling the CAU and using TMR voting on its output. Although a TMR configuration ultimately becomes less reliable than a single unit, this does not occur until about 7 times the mean life of the individual unit. Since the voter unit has relatively few components, its mean life is very long and therefore TMR is to advantage here. The type of system discussed and illustrated in the figures is known as a closed system in that at the beginning all equipment is present with no additions or repairs possible thereafter.

In analyzing system reliability, two important points for consideration are the rate of degradation of standby units and the switch reliability.

Kletschy⁵ shows that the mean life of a closed modular set cannot be increased significantly when active and standby failure rates are assumed equal. Therefore, it is necessary to obtain a value for this ratio. Little direct data, however, is available for "d", the ratio of standby failure rate to active failure rate. However, data reported by Nerber⁶ indicate that d is considerably less than unity. Nerber analyzed field data for

over 100 transistorized guidance computers. From this data a maximum value of d can be inferred to be about 0.33. A more recent analysis of Minuteman II computer failures by Watson⁷ shows that the expected value of d for integrated circuits is 0.55. A lower bound for this ratio appears to be 0.12. Extrapolation indicates that the ratio will decrease as more data is gathered. For future missions, it seems conservative to assume that the ratio will lie below 0.5 with 0.1 reasonably attainable. Though this ratio may further decrease as more is learned, the greatest significance for the closed module set reliability is effected with a d of the order of 0.1.

The reliability model must also consider the effect of the switches. Though the switch size is held at a minimum (typically 8 data bits + 5 control) its effect upon long term reliability if not properly treated can be great. For instance, the reliability of a single cross-point (with a normalized failure rate, $\lambda_s = 2 \times 10^{-7}$) is .991, for a 3×3 switch .925, and the probability that 3 crosspoints out of the 3×3 switch will be operational is $1 \cdot 10^{-10}$. Thus, for long term system reliability of the order of .99 proper treatment of the switch reliability is imperative.

The following paragraphs describe the mathematical model developed to accommodate the above factors

Reliability model

The block diagram of Figure 2 will serve as a framework for the mathematical model. The mathematical model is somewhat more general than Figure 2 in that there may be more than four levels (a level consists of all modules of one type) of modules with the j -th level having N_j modules out of which L_j are required operational with $N_j - L_j$ being in standby at the beginning. The computer system is operational as long as L_j modules are operational at the j -th level for all j 's with unfailed switch capability for interconnection of the L_j 's from level to level for all levels and the CAU is operational.

The switch between each level allows connection of any module at one level to any module at the other level. Further, the switch is designed so that independent failures may occur in the switch such that certain switching connections are disabled without affecting other connections (all other type switch failures can be associated with a module). Thus, one failure in a switch may make connection of module 1 of one level to module 2 of another level impossible without affecting the connectability of module 1 to

module 1, module 2 to module 2, or module 2 to module 1. Figure 3 illustrates a typical switch.

It is desired to find the reliability of the total modular computer system at any instance of time. L modules must operate at each level as well as the switches to interconnect them. The problem is approached by first finding the reliability at each level then iterating from level to level including at each step the switch reliability.

If switching redundancy is applied to a module level with L replicas operating and $N-L$ in standby the reliability according to Kletsky (Reference 5) can be given as an inverse Laplace transform.

$$R_R = 1 - \mathcal{L}^{-1} \left\{ \frac{1}{S} \prod_{K=R-L}^{N-L} \frac{(L + Kd)\lambda}{S + (L + Kd)\lambda} \right\} \quad (1)$$

where R is the number of modules operational (unfailed) for $L \leq R \leq N$ and λ is the active failure rate of the module while d is the ratio of standby to active failure rates.

The Reliability can be reduced to:

$$R_R(t) = \sum_{K=R-L}^{N-L} \left\{ e^{-(L+Kd)\lambda t} \left(\prod_{\substack{h=R-L \\ h \neq K}}^{N-L} \frac{L/d + h}{h - K} \right) \right\} \quad (2)$$

Since failure is characterized by independent random variables the probability ($P_R(t)$) of exactly $N-R+1$ failures (R operational) is equal to:

$$P_R(t) = R_R(t) - R_{R+1}(t)$$

taking the difference and combining terms leads to:

$$P_R(t) = \sum_{\substack{P=R-L \\ P \neq R-L}}^{N-L} \left(\frac{(L/d + P)}{R-L-P} \right) e^{-(L+Kd)\lambda t} \prod_{\substack{h=R-L+1 \\ h \neq K}}^{N-L} \left(\frac{L/d + h}{h - K} \right) \quad (3)$$

To model the effect of the switches a recursive procedure is used beginning at Level 1, then Level 2 and onto the highest level.

Let $P_{RS}^{(j)}(t)$ designate the probability that at time t exactly R modules are operational at the j -th stage and exactly S ($S \leq R$) of them may be reached from Level 1 through a path of operational modules and sub-switches at the lower levels (Reference 8). Since $R \leq N$ the states of the process that yield at least one com-

puter's worth of capability (for the higher level) correspond to:

$$(R, S) = (1, 1), (2, 1), (2, 2), (3, 1), (3, 2), (3, 3), \dots, \\ (N, 1), (N, 2), (N, 3), \dots, (N, N)$$

with reference to the switch of Figure 3, the equation governing transition between the various levels is given by:

$$P_{RS}^{(j)}(t) = P_{R^{(j)}}(t) \binom{R}{S} \sum P_{UV}^{(j-1)}(t) \cdot W_{RSV}^{(j-1)} \quad (4)$$

$$L_{j-1} \leq V \leq U \leq N_{j-1}$$

where $W_{RSV}^{(j-1)}$ is the probability that exactly S modules at level j are connectable through the switch from V modules at level $j-1$. For exponential failure:

$$W_{RSV}^{(j-1)} = (1 - (1 - e^{-\lambda'(j-1)t})^V)^S (1 - e^{-\lambda'(j-1)t})^{V(R-S)} \quad (5)$$

where $\lambda'(j-1)$ is the cross point failure rate of the switch at the $j-1$ st level.

Substituting Equations (3) and (5) into Equation (4) yields the expression for the probability

$$P_{RS}^{(j)}(t) = \sum_{\substack{p = K = R - L_{(j)} \\ p \neq R - L_{(j)}}}^{N_j - L_j} \left\{ \frac{L_{(j)}/d + p}{R - L_{(j)} - p} \right. \\ \left. e^{-(L_{(j)} + Kd)(j)t} \prod_{\substack{h = R - L_{(j)} + 1 \\ h \neq K}}^{N_{(j)} - L_{(j)}} \frac{L/d + h}{h - K} \right\} \quad (6)$$

$$\left\{ \binom{R}{S} \sum P_{UV}^{(j-1)} (1 - (1 - e^{-\lambda'(j-1)t})^V)^S \cdot \right. \\ \left. L_{(j-1)} \leq V \leq U \leq N_{(j-1)} \right. \\ \left. (1 - e^{-\lambda'(j-1)t})^{V(R-S)} \right\}$$

where the number of modules $N_{(j)}$, those required operation simultaneously, $L_{(j)}$, and a module failure rate $r_{(j)}$ are variable from level to level.

Thus, the $P^{(j)}(t)$ vectors are obtained recursively starting first from $P^{(1)}(t)$ which is given component-wise by:

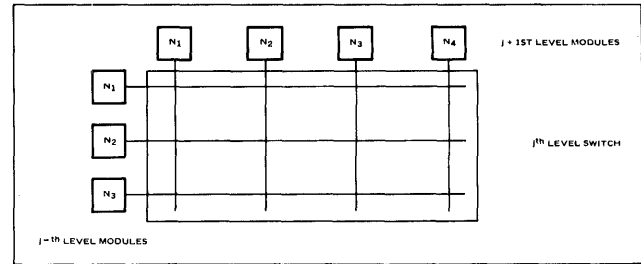


Figure 3—Module-switch-module relationships

$$P_{RS}^{(1)}(t) = \sum_{\substack{p = K = R - L_1 \\ p \neq R - L_1}}^{N_1 - L_1} \frac{(L/d + p)}{R - L - p} \\ e^{-(L_1 + Kd)\lambda_1 t} \prod_{\substack{h = R - L_1 + 1 \\ h \neq K}}^{N_1 - L_1} \frac{L_1/d + h}{h - K} \quad \text{if } S = R \\ = 0 \quad \text{if } S \neq R \quad (7)$$

The total system reliability then including modules, switches, and CAU is given by:

$$R(t) = R_{CAU} \prod P_{RS}^{(m)}(t) \quad (8) \\ L_m \leq S \leq R \leq N_m$$

where m is the highest module level and the reliability of the CAU is given by R_{CAU} .

$R(t)$ then gives the probability that at time, t , at least one string of modules (L_j at each level) is connectable and operational and further that the CAU can switch modules in and out of the operating string when needed.

Two assumptions implicit in this derivation should be noted. First that in assuming a strict exponential failure distribution for the modules wherein a module was assumed to fail at a rate λ if in the active state and λd if in the passive state, no allowance was made for failures which might occur by change of state transitions. No algorithm exists for switching modules on or off to form an operational string which guarantees full utilization of the remaining system without requiring that some modules be placed in successive modes of active and standby states. It is therefore assumed that module failure is not influenced by its history of active and standby state transition. To solve the problem otherwise becomes extremely difficult.

The second assumption is that all crosspoints of a switch are assumed to be in the active state whether or not the corresponding modules are presently connected, and therefore the failure rate of crosspoints fail only at the active rate (no standby rate).

The numerical calculation may be facilitated by re-writing Equation (3) as:

$$P_R(T) = \binom{L/d + N - L}{N - R}^* \sum_{K=R-L}^{N-D} (-1)^{-R+L+K} \left(\frac{N-R}{N-L-K} \right) e^{-(L+Kd)T} \quad (9)$$

Then expanding and collecting factors

$$P_R(T) = \binom{L/d + N - L}{N - R}^* e^{-(L+(R-L)d)T} (A_0 - e^{-dT}(A_1 - e^{-dT} \dots (A_n - e^{-dT}) \dots))$$

where

$$A_n = \binom{N-R}{n} \text{ for } 0 \leq n \leq N-R-1 \quad (10)$$

Equation (8) requires computer solution. In order to achieve a better intuitive grasp for the components of reliability of a modular system and to help in the initial selection of architectural organization with potentially high reliability over lengthy missions, a first estimation of the reliability of a system may be made with the aid of the graphs of Figure 4. The figure shows the reliability of modules with various degrees of redundancy against a normalized time scale. Notice the change of scale. In the first estimation of the reliability the effect of the switches can be neglected so that reliability of the system:

$$R_s = R_{CAU} \prod_{j=0}^m R_j \quad (11)$$

where R_j is the reliability of the modules at the j -th level for all j 's. The R_j 's may be found on the ordinate axis of the graph of Figure 4 when the normalized time has been computed.

* Use generalized factorial function i.e. $(n)! = \Gamma(n+1)$

To find the reliability of a particular closed module set first compute its equivalent units of normalized time which are equal to the product of number of gates, reliability of a gate, and time; then read the corresponding reliability from the graphs. Use Equation (11) for approximate system reliability. This procedure not only allows one to work out a reasonable configuration but also indicates what the basic gate reliability must be to attain the required reliability with a feasible organization.

To illustrate the procedure by way of example, consider the configuration of Figure 2. Each level can be characterized by three parameters: N the number of modules, L those operating simultaneously and T normalized time. The system then can be summarized as:

	CAU	AU	CU	MU	I/O
N	1	3	3	3	3
L	1	1	1	1	1
T	2.1	1.49	3.54	.85	.88

To compute normalized, gate count is obtained from Table I, the failure rate is assumed to be $\lambda = 10^{-8}$ failures per hour per gate, normalized time is computed for five years and standby failure rate is 1/10 of active failure rate ($d = .1$). Module set reliabilities are read from the appropriate curves of Figure 4 as follows:

	CAU	AU	CU	MU	I/O
T	2.1	1.49	3.54	.85	.88
R	.12	.76	.24	.92	.91

Then system reliability is:

$$R_s = (.12) (.76) (.24) (.92) (.93) = .018$$

Changes must obviously be made for a reasonable reliability. In Figure 5 a step toward higher reliability is demonstrated through several types of changes. The AU and CU are each divided into interchangeable halves and the AU provided with four spare halves while the CU has five. One unit each is added to the I/O and MU units. The CAU has been triplicated and at the same time reduced to one half its former size. The system summary is:

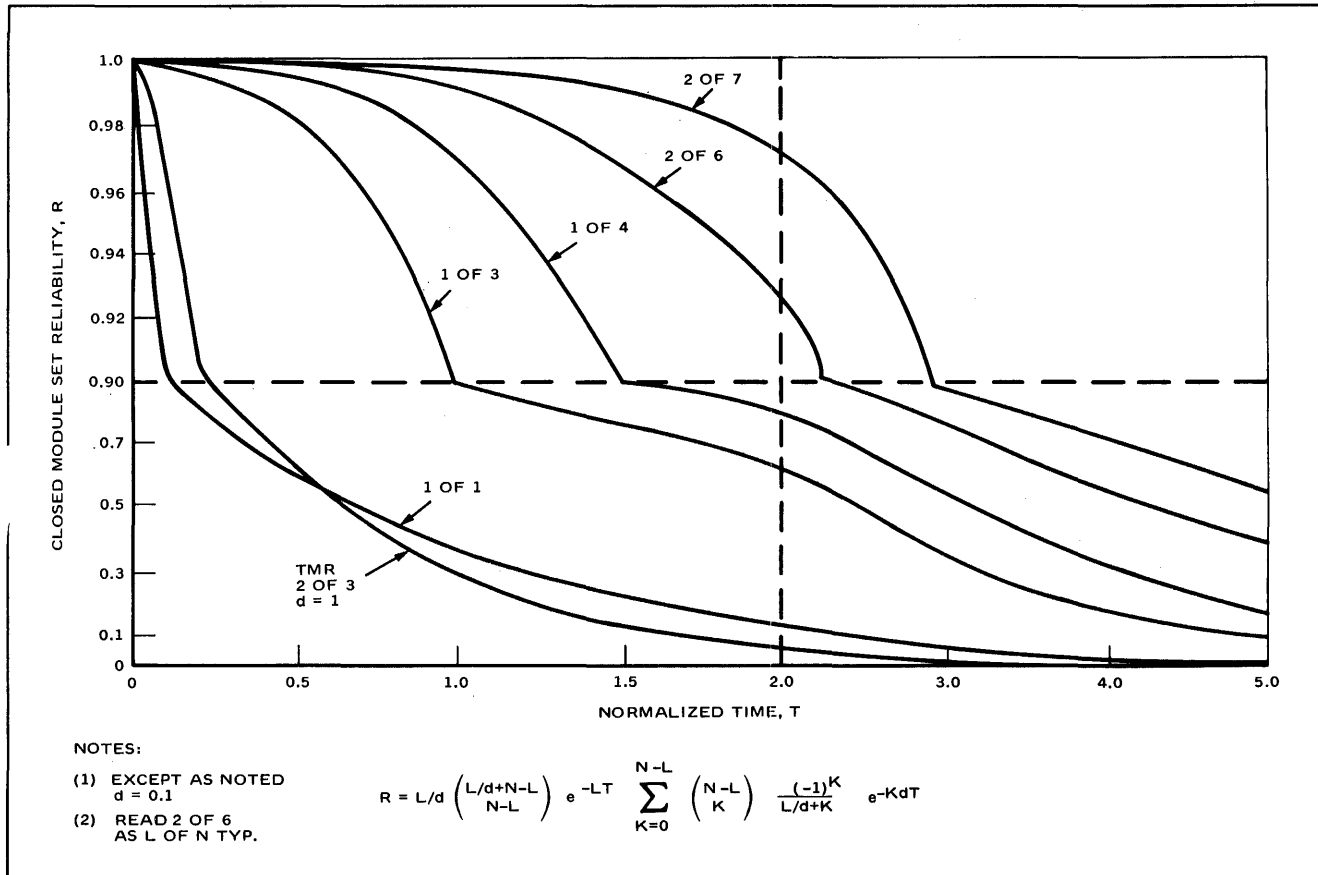


Figure 4—Long term reliability curves

	CAU	AU	CU	MU	I/O
N	3	6	7	4	4
L	1	2	2	1	1
T	1.05	1.49	3.54	.85	.88

Reading the graphs for module set reliability:

	CAU	AU	CU	MU	I/O
T	1.05	1.44	3.54	.85	.88
R	.89	.97	.8	.98	.98

Thus, $R_s = .67$ — a significant gain in system reliability, though additional steps must yet be made to reach the desired reliability. When a potential con-

figuration is reached one must determine the additional gates added to each module by virtue of the new configuration and then calculate a corrected system reliability. Usually one must iterate through several configurations many times to reach the desired reliability with a minimal gate count. At this point Equation (8) may be used for a more accurate reliability value.

CONCLUSIONS

A method of estimating long term reliability of modular computers has been presented and two sample cases examined. In the second example 240 percent additional hardware was used to improve five year predicted reliability from .018 to .67. To this must be added the additional switches to accommodate the increased modules (from 13 in first example to 24 in second). To obtain a reliability of the order of .99 for a five year mission perhaps the additional hardware necessary

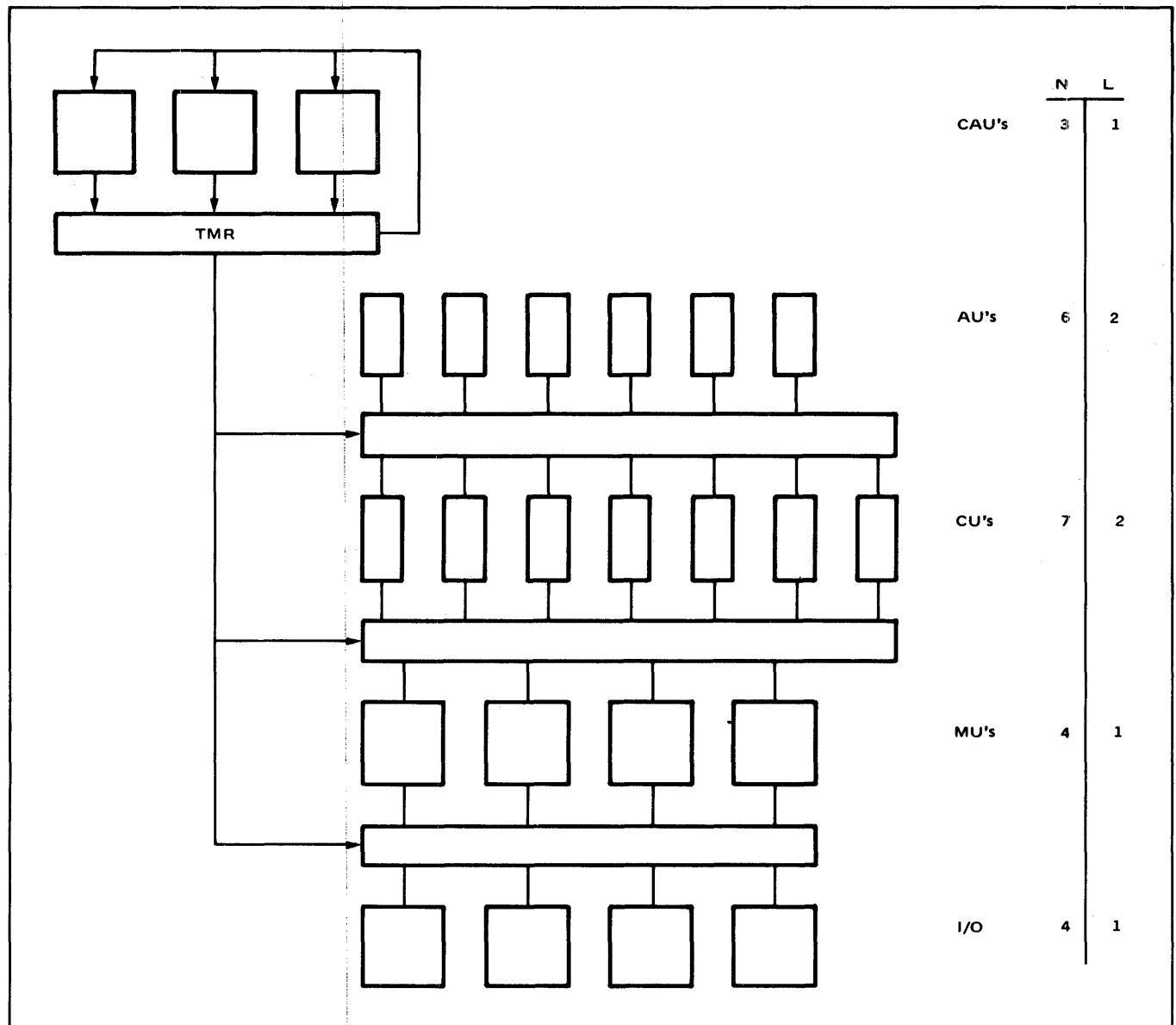


Figure 5—Multi-module modular computer

would amount to as much as four times that required for the actual computing. Gate failure rates used in the examples are for present day high quality IC's. If the basic gate reliability could be increased by a factor of ten this total additional hardware could be approximately halved.

The modular approach with standby modules appears capable of servicing long missions with feasible costs.

ACKNOWLEDGMENT

The authors express their appreciation to Mr. Jack

L. Bricker of Hughes Aircraft Company for his effort and guidance in developing the mathematical model.

REFERENCES

- 1 J J PARISER H E MAURER
Modular computer implementation with LSI
In these proceedings
- 2 F D ERWIN J F Mc KEVITT
Characters—Universal architecture for LSI
In these proceedings
- 3 R A SHORT
The attainment of reliable digital systems through the use of

TABLE I—Two column component breakdown (approximate)
modular computer breadboard (separate arithmetic & control modules)

MODULE	GATES/MODULE	%	IC/MODULE	%
CAU	4800	24	1440	25
Switches	180	1	55	1
I/O	2000	10	495	9
Memory	1950	10	495	9
Control	8100	40	2175	39
Arithmetic	3400	15	975	17
TOTAL	20430	100	5635	100
ALTERNATE APPROACH (COMBINED ARITHMETIC & CONTROL MODULES)				
CAU	4800	25	1440	27
Switches	180	1	55	1
I/O	2000	11	495	10
Memory	2000	11	495	10
Processor	9800	52	2700	52
TOTAL	18780	100	5185	100

redundancy—A survey

Computer Group News March 1968

4 BERSOFF HOPE TUNG

Modular computer research

To be published

5 E J KLETSKY

Upper bounds on mean life of self-repairing systems

IRE Trans on Reliability and Quality Control Oct 1962
43-48

6 P O NERBER

Power-off time impact on reliability estimates

IEEE Internat Conv Record Part 10 March 22-26 1965
N Y 1-8

7 L K DAVIS G A WATSON T G SCHAIRER

Advanced computer dormant reliability study, Final Report

Autonetics Div of No America Rockwell Corp Oct 14 1967

8 J L BRICKER

*Reliability studies of the NASA deep space computer and the
H-4400 computer*

To be published

