General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

ARC

CR-86320

Loan
Return
ASAP to
ERC
attention Paul Corey
Code ANP

Paul Corey
Code ANP
ERC

FROM : Documentation Services

SUBJECT: Transmittal of High Number Contractor Report
(CR-86320)

The following report is submitted for announcement and
abstracting in STAR:

FAULT DIAGNOSIS OF OPERATIONAL SYNCHRONOUS
DIGITAL SYSTEMS.

Subject report was prepared under contract NAS 12-692 by the
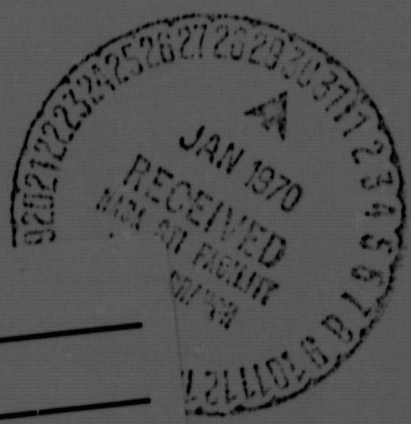University of Missouri - Columbia.

The technical monitor who recommends dissemination as a
high-number contractor report is Edward Sarkisian.

Dolores G. Crosby

Enclosures:
1. Subject Report (1 copy)
2. Document Release Form

cc: DLM/H. Farmer
    DAT/F. Hills
    AT/C. Hadge

FAULT DIAGNOSIS OF OPERATIONAL SYNCHRONOUS DIGITAL SYSTEMS*

MICHAEL J. DEVANEY AND GEORGE W. ZOBRIST +

The problem consists of diagnosing faults on operational
synchronous digital systems. The paper presents an
original approach partitioning the fault diagnosis problem
into fault detection and fault location enabling the
detection of single and distinguishable multiple faults
and the location of these faults down to their defective
module or package in order that effective corrective action
can be taken. As the anticipated application of this
approach is in aerospace systems, effort has been exerted
to minimize computer time and storage requirements so that
it may operate effectively on a non-dedicated computer in
a time-shared environment. The effectiveness of the approach
is demonstrated by its application to a Boolean model of
the Gemini's Electronic Timer.

To the authors' knowledge the paper offers a new approach
in spaceborne systems and the material presented has not
been published elsewhere.

INTRODUCTION

Recent advances in the design of digital systems have resulted in ever-increasing complexity in such systems. Concomitant with this rise in complexity is the growing demand for extending the operational life of these systems. The combination of these factors focuses increasing emphasis on the problem of equipment maintainability.[1] A requisite condition for effective equipment maintainability and the particular phase with which this investigation is concerned is the development of an efficient fault diagnosis technique. The technique for error detection and fault diagnosis described is directed to isolating logic failures in operating synchronous digital systems. It is anticipated that this approach is to be utilized in an environment which can tolerate only a very small amount of system down time. Typical applications for the method include guidance computers, aircraft collision avoidance systems, navigational time reference systems, etc.

The technique introduces a Model Assisted approach to Bi-Modular Redundancy providing continous error detection, fault diagnosis to the module level, and a self-repair capability, by means of which the system is automatically reconfigured to bypass the failure and restore operation until the defective module can be replaced or repaired. The theoretical basis for the approach is presented and an algorithm is developed for generating an optimal sequence of diagnostic tests. The study concludes by describing the simulation of Model Assisted BMR as applied to the Electronic Timer of the Gemini's Time Reference System.

MODEL ASSISTED BI-MODULAR REDUNDANCY

The Model Assisted Bi-Modular Redundant approach to fault diagnosis is in-
troduced by considering an elementary Parallel Redundant System, identifying
its shortcomings, and demonstrating how these shortcomings are overcome
in a Model Assisted Bi-Modular Redundant System.
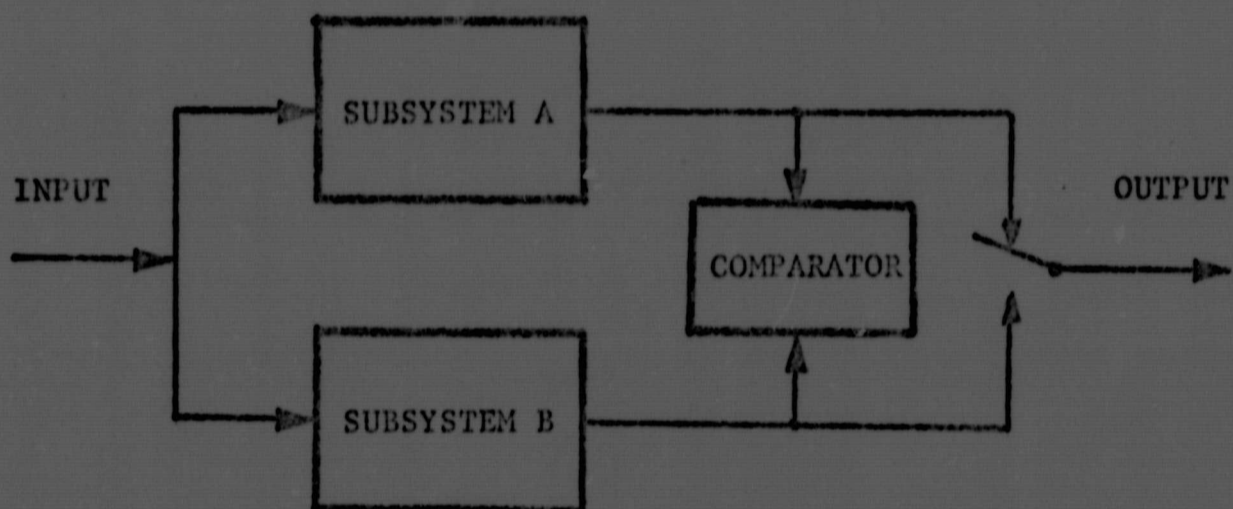
Parallel Redundant Example

Figure 1

Parallel Redundant System

A Parallel Redundant System is depicted in Fig. 1. A comparator is used
to monitor the outputs of subsystems A and B which possess a common input.
The system also contains a switching element capable of selecting the output
of either subsystem. The switch as indicated is selecting A as the primary
subsystem while B functions as a reference. The comparator provides
error detection by computing the Boolean difference of the subsystem out-
puts and thereby indicates disagreement when a difference is observed.
When this condition occurs system operation is interrupted and both subsys-

tems are subjected to a battery of diagnostic tests. If these tests are successful in localizing the fault to subsystem A, the output switch is thrown to B and this system takes over the role of the primary system, while A is repaired or replaced. The converse situation occurs if the diagnostics indicate subsystem B is faulty.

Several shortcomings are observable in this Parallel Redundant approach to fault diagnosis. Paramount among these is the dependence of the method on the set of diagnostic tests. The difficulty encountered in developing efficient diagnostic test sequences for the sequential circuitry prevalent in most digital systems can constitute a major handicap. While adequate methods have been developed for the test synthesis for strictly combinational circuitry (provided these circuits contain no redundancy), there is no simple straight-forward method for developing the diagnostic tests for sequential systems.[2] The methods which have been described in the literature for devising tests for sequential circuitry are usually based upon a single fault hypothesis of the logical node "stuck-at-one" or "stuck-at-zero" variety.[3,4] These methods, while useful in an inspection environment, very often result in such lengthy test sequences as to render them impractical in an operational environment.

An additional shortcoming of this elementary system results from the fact that a faulty subsystem having been diagnosed, the entire system is deprived of its error detecting capability until this subsystem is replaced or repaired.
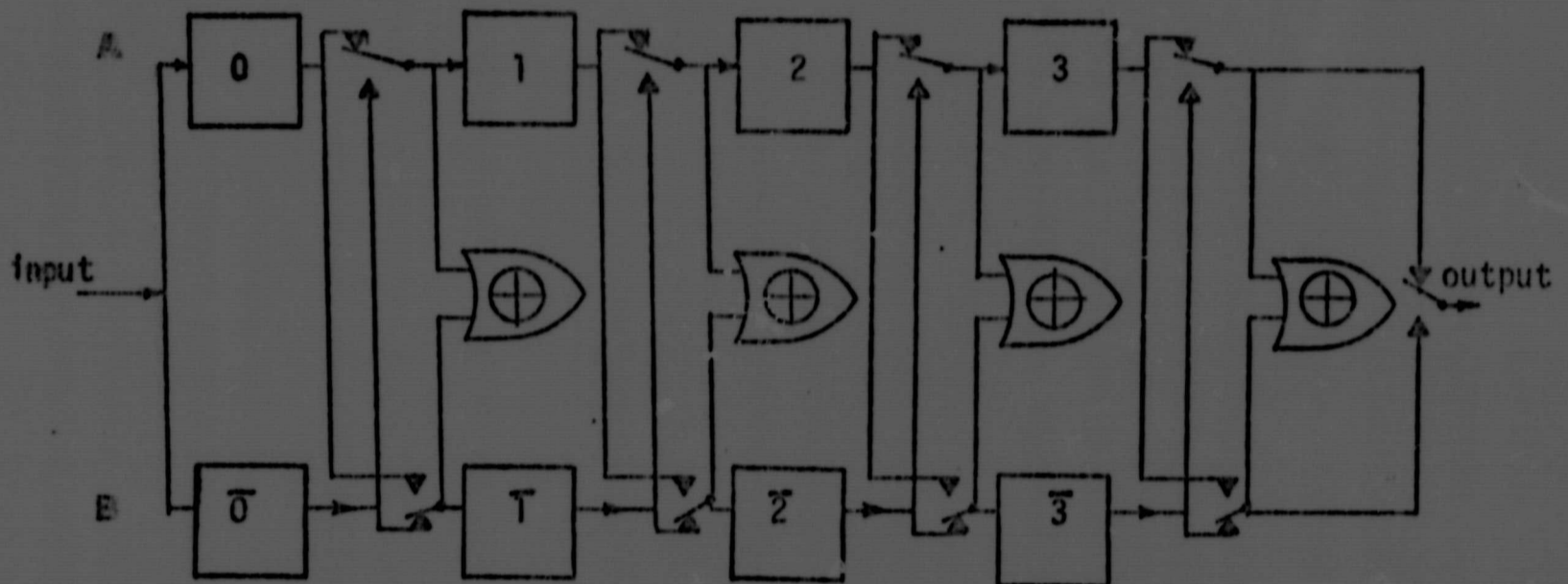
Bi-Modular Redundancy

Figure 2   Bi-Modular Redundant System

This latter handicap can be alleviated by decreasing the level of re-
dundancy from the subsystem level to the module level. Figure 2 depicts
a Bi-Modular Redundant System. (Although the figure suggests series con-
nected subsystems composed of single input single output modules neither the
BMR system nor the method to be developed for fault diagnosis are restrict-
ed to this type of module or this connection topology.) All intermodular
connections in this system traverse steering networks which function as
S.P.D.T. switches. Exclusive-or gates have been located at the inputs
to the modules to provide disagreement detection. When a fault is detect-
ed and diagnosed to a specific module the switches on all outputs from
this module are placed in their alternate position. This action isolates
the defective module and allows its counterpart to perform for both subsys-
tems while this modu   is out of service. Locating the error-detecting
logic across the input side of the switches inhibits the detector imme-
diately following a faulty module allowing the remainder of the error-

4

detecting logic to remain effective. If a fault is detected in the switching logic or in the comparison logic it can be corrected in the same manner as the intramodular faults.

Sequential Machine Theory Provides Foundation for MASMR

The theoretical basis for utilizing a Boolean model of the Subsystem to assist in failure diagnosis is couched in the fundamental theory of synchronous sequential machines.
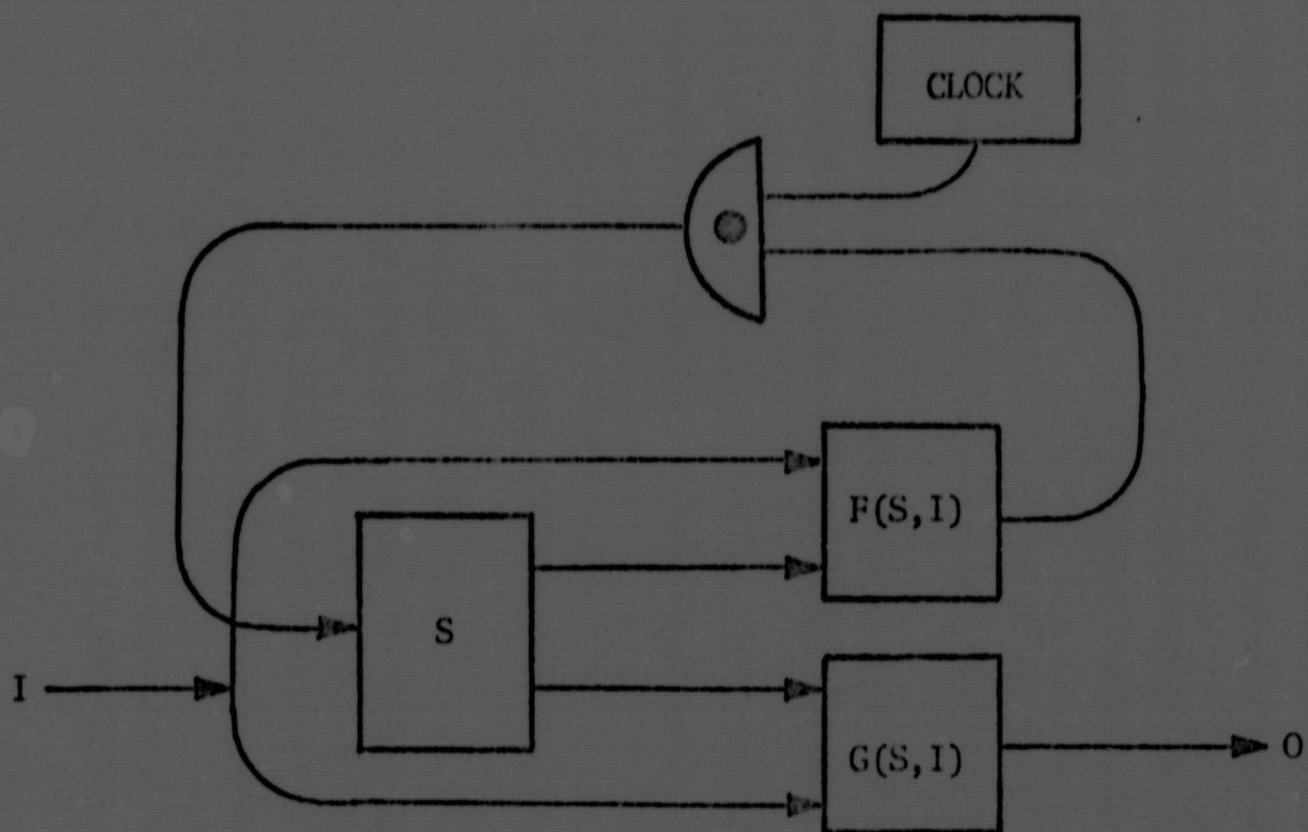
Figure 3

Block Diagram of General Sequential System

Any deterministic synchronous sequential machine may be depicted functionally by the block diagram of Figure 3.

$$S(t) = [s_1(t), s_2(t),...,s_n(t)], \tag{1}$$

is the state vector of the general machine. This n dimensional vector identifies the status of the N internal memory elements within the machine and as such may be viewed as an N bit register.

$$I(t) = [i_1(t), i_2(t),...,i_m(t)] \tag{2}$$

and

$$O(t) = [o_1(t), o_2(t),...,o_p(t)] \tag{3}$$

are the m dimensional input vector and the p dimensional output vector respectively. $F(S(t), I(t)$ is a n-vector valued function defined over $S \times I$, while, $G(S(t), I(t))$ is a p-vector valued function defined over this same product space. These functions can be realized by strictly combinational logic. Since the occurrence of transitions in synchronous sequential machines are restricted to clock pulses, the explicit time difference equation for this machine may be stated as:

$$S(t + T) = F(S(t), I(t))$$

and $\tag{4}$

$$O(t) = G(S(t), I(t))$$

These equations are reminiscent of the state equation characterization for continous systems. If the first of these equations is operated on by the backward shifting operator so that the dependent variables coincide in time the equations become:

$$S(t) = F(S(t-T), I(t-T))$$

and $\tag{5}$

$$O(t) = G(S(t), I(t))$$

Thus if the status of the system during the previous period is known and its present input is available then the state of its internal memory elements and that of its outputs are ascertainable.
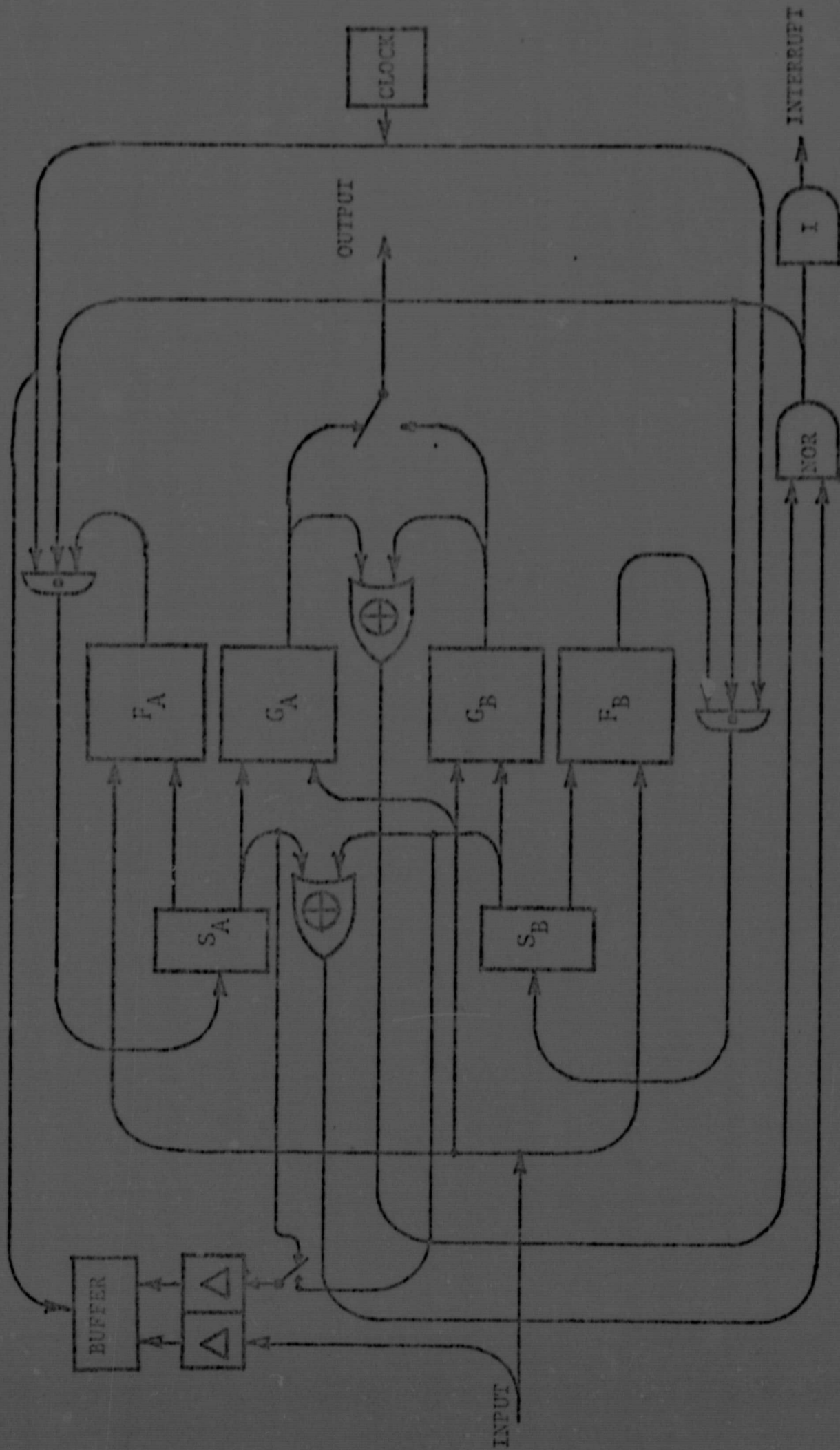
Figure 4

Functional Diagram of Synchronous Bi-Modular Redundant System

Operational Characteristics of the MABMR System

The Fault Diagnosis capability of a synchronous Bi-modular Redundant
System is depicted in the block diagram of Figure 4. The State and Output
vectors from both subsystems are compared each clock cycle. If they coin-
cide the State vector and the current Input vector are delayed by a single
clock period and stored in a M + N bit buffer register. The system contin-
ues to function in this manner until the error detection logic detects
disagreement in the subsystem State and or Output vectors. When this sit-
uation occurs the clock is inhibited freezing the system in its current
state and an interrupt is generated to a small general purpose computer.

This computer on receiving the interrupt loads the Boolean model of the
interrupting system. This model consists of a sequence of logical equa-
tions which implement the vector valued functions $F(S(t-T),I(t-T))$ and
$G(S(t),I(t))$. With the model loaded, the computer retrieves the contents of
the BMR Buffer register containing the previous State and Input vectors and
evaluates the function $F(S(t-T),I(t-T))$ to obtain $S_m(t)$. The current input is
now retrieved and the function $G(S(t),I(t))$ evaluated for $O_m(t)$. The computer
should now contain the State and Output vectors for the faultless system.

The comparison of the model generated State and Output vectors ($S_m(t)$ and
$O_m(t)$) with the corresponding vectors from subsystems A and B will indi-
cate the defective subsystem if all the faulty modules producing the error
condition reside within a single subsystem. This condition is satisfied for
all single defective module situations. This first comparison will be con-
sidered test zero (T_o). Assuming the integrity of the model generated

8

res,onse (S_m and O_m), this initial test can have the following four possible outcomes:

I.	$A = M = B$
II.	$A = M \neq B$
III.	$A \neq M = B$
IV.	$A \neq M \neq B$

If condition I occurs the defect is diagnosable to the error detecting logic. Condition II and III isolate the defective modules to subsystems B and A respectively, while condition IV reveals defective modules in both subsystems.

Since the computer utilized in a MABMR system is required only after a system error is detected, only a small portion of the computer's time would be dedicated to failure diagnosis. Thus, this computer could be performing a number of other functions perhaps in a multiprogramming or time shared environment until interrupted by the error detection logic of the MABMR system. A priority interrupt scheme would warrant consideration where two or more of these systems are being serviced. If mission requirements are insufficient to justify the presence of this computer onboard, its capability could be provided via telemetry.

While an appropriate subsystem could be selected under conditions I, II, or III by masking the interrupt, selecting the subsystem, and enabling the clock, a small amount of additional system time will isolate the detected faults to their respective modules as indicated in the next section.

9

Fault Diagnosis and Fault Correction in a MABMR

The diagnosis is performed by interchanging corresponding modules from subsystems A and B by actuating pairs of the intermodular switches previously described. The circumstances under which the faulty condition was originally detected are then duplicated. The system diagnosis and restoration procedes according to the following iterative algorithm.

1. Interchange corresponding modules from subsystems A and B as specified by the Test Vector for this iteration.

2. Restore system inputs and memory elements to their status for the period immediately prior to original error detection.

3. Single-cycle the system clock.

4. Apply inputs occurring during the original error detection period.

5. Compare the responces of both subsystems with the previously computed response.

6. Return to 1 if the status of any module remains questionable.

7. Actuate switches to isolate faulty modules.

8. Restore system operation.

As indicated in step 3 the minimum time required for each test is slightly longer than the cycle time of the system clock. Because of this factor the algorithm will usually converge quite rapidly yielding all the detectable faulty modules within the system. The convergence criteria are first, that there exist at least one of the 2^N configurations of the subsystem which is non-defective, and second, that the proper set of tests is utilized

10

in step 1. A method for determining the optimum sequence of these tests
is developed in the next section.

Once the defective modules have been isolated and the system restored to
operation, error detection and fault diagnosis continue o'   on that portion
of the original system which remains bi-modularly redundant until these
defective modules have been replaced or repaired. If the defective modules
are not replaced or repaired as they occur, and as more and more failures
occur, the fault diagnosis algorithm will gracefully degrade until either
it will no longer be significantly effective in detecting errors, or there
will no longer exist an effective configuration of the system.

Optimal Test Sequences for MABMR Systems

The diagnostic test sequence resulting from the algorithm developed in this
section is optimal in the sense that it yields the minimum number of tests
necessary to distinguish fault conditions of a given class. The term
fault condition denotes the particular combination of defective modules which
produced the detected error, while the term _fault order_ refers to the number
of these modules. All fault conditions of a given order are assumed to have
a nearly uniform probability of occurrence. Although, the algorithm can be
adapted to handle particular situations where there is a great disparity
in these values. An additional assumption is made that the lower the fault
order, the more probable the individual fault conditions of this order.
Recalling that the error detecting logic compares the two subsystems at each
clock cycle, this becomes a relatively safe assumption.

The algorithm is iterative in nature. It first selects the minimum number of tests necessary to distinguish all first order fault conditions. These are the single defective module situations. The algorithm then on the basis of previously selected tests selects the minimum number of additional tests to distinguish all correctable fault conditions of the second order. If a BMR system contains 2N modules, then the highest order fault conditions which this system can tolerate are of order N. If the algorithm is allowed to continue considering correctable fault conditions of successively higher orders until it completes the Nth order conditions, the resulting test sequence will be the shortest able to distinguish all correctable fault conditions. If the algorithm is terminated prematurely, then it will provide the shortest test sequence able to distinguish all fault conditions up to the highest order completed. Under these circumstances the algorithm will generate the minimum number of tests necessary to distinguish all fault conditions whose probability exceeds some lower bound.

The particular configuration of the BMR system being examined during any test is determined by the test vector. This vector is referrenced to the status of the modules prior to error detection. The Jth test vector in the test sequence is defined as follows:

$$T_j = [t_{j,0}, t_{j,1}, \ldots, t_{j,N-1}] \qquad (6)$$

Where the value of the component for level k ($t_{j,k}$) is zero if module k, which under this test configuration is an element of subsystem A, originated in A and module $\bar{k}$, now residing in B, originated in B. Otherwise $t_{j,k}$ is one implying k and $\bar{k}$ are rotated from their original status.

12

Since after the initial comparison the detected defect is resolved to either the error detecting logic or the remainder of the system, the possible outcomes for further tests are conditions II, III, or IV. These test results can be recorded in ternary; "0" means the error condition stems from subsystem A alone (III), "1" implies that it stems from B alone (II), while "2" indicates that both subsystems are faulty (IV). The test data ("0's","1's", and "2's") may be arranged in a matrix form. The matrix $D_k = [d_{i,j}]_{q,r}$ is called the Data Matrix for fault conditions of order K. This matrix has one row for each fault condition of this order $f_i (1 \leq i \leq q)$, and a column for each test $t_j (1 \leq j \leq r)$. The element $d_{i,j}$ of D is zero, if fault condition f_i under test t_j resides in system A alone; one, if it resides in B alone; or two, if it resides in both. The rows of D are called fault patterns while the columns of D are called test patterns.

The matrix D_k is obtained by inserting each fault condition of a given order into a binary model of the system, executing each test, and recording the the results as previously described. The binary model for a BMR system of 2N modules consists of two N bit words corresponding to the A and B subsystems. These words are initially zero. The fault condition is inserted by setting corresponding bits within each word. The test is executed by interchanging the appropriate bits from word to word and then testing each word against zero. A non-zero word denotes the presence of a faulty module within the corresponding subsystem. The result is recorded in ternary for the particular element of D_k.

Since the complement of a test vector merely exchanges the status of the subsystems and as such possesses no additional fault distinguishing capability,

13

	0	1	2	3
0	0	0	0	0
1	0	0	0	1
2	0	1	0	0
3	0	1	1	1
$\bar{3}$	1	0	0	0
$\bar{2}$	1	0	1	1
$\bar{1}$	1	1	1	0
$\bar{0}$	1	1	1	1

Fig. 5(a)

T_0 [0 0 0 0]

T_1 [0 0 1 1]

T_2 [0 0 0 1]

T_3 [0 1 0 1]

Fig. 5(b)

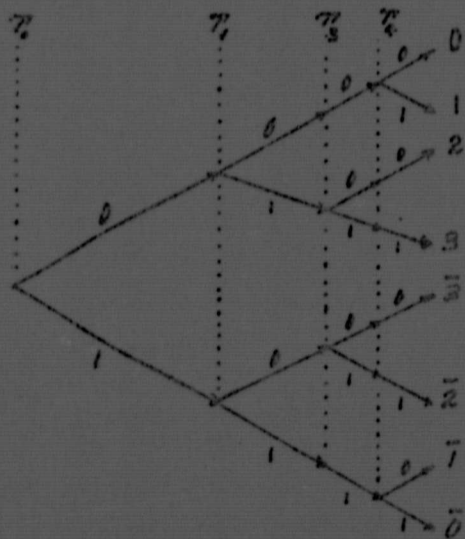Fig. 5(c)

$S_i \backslash t_j$	0	1	2	3
01	0	0	2	2
23	0	1	2	2
02	0	2	0	2
13	0	2	1	2
03	0	2	2	0
12	0	2	2	1
$\overline{23}$	1	0	2	2
$\overline{01}$	1	1	2	2
$\overline{13}$	1	2	0	2
$\overline{02}$	1	2	1	2
$\overline{12}$	1	2	2	0
$\overline{03}$	1	2	2	1
$0\bar{3}$	2	0	0	2
$1\bar{2}$	2	0	1	2
$0\bar{2}$	2	0	2	0
$1\bar{3}$	2	0	2	1
$2\bar{1}$	2	1	0	2
$3\bar{0}$	2	1	1	2
$3\bar{1}$	2	1	2	0
$2\bar{0}$	2	1	2	1
$0\bar{1}$	2	2	0	0
$2\bar{3}$	2	2	0	1
$3\bar{2}$	2	2	1	0
$1\bar{0}$	2	2	1	1

Fig. 5(d)

the complement of a selected test is redundant. To avoid this redundancy
the modules 0 and $\bar{0}$ are restricted to subsystems A and B respectively as
indicated in figure 2. Thus, to obtain the global optimum, 2^{N-1} test vectors
must be considered. If less than this number are evaluated, a local optimum
is achieved among those considered. The objective then is to select the
minimum number of additional tests such that every fault pattern is disting-
uishable from every other fault pattern of this order.

The matrix D may be interpreted graphically in terms of a rooted directed
tree called a decision tree. Figure 5 (a) contains a submatrix (three of the
eight test patterns have been excluded) of the D_1 matrix of the BMR system
of figure 2. The particular test vectors associated with each test pattern
appear in 5(b). The decision tree associated with this matrix is depicted
in 5(c). Each vertex of the tree represents a decision. The edges emanating
from these vertices have transmittances corresponding to the possible outcomes.

In decision trees for distinguishing fault conditions of order two or greater,
the ternary decision rule allows as many as three edges to exit a given vertex.
However, as indicated in figure 5, first order decision trees can provide
at most binary decision vertices since condition IV cannot occur for fault
conditions of this order. The set of edges ordered from the root to a given
vertex is referred to as its branch, while the number of edges in the branch
determines the vertex decision level. All vertices of decision level j are
associated with Test Vector T_j and the edges leaving these vertices determine
Test Pattern t_j.

Each branch of the decision tree may be expressed as a subset of the data

15

matrix D as:

$$B_e \, [t_{io}, t_{i1}, \ldots, t_{is-1}, \ldots, t_{ir}] \qquad (7)$$

The columns of this matrix are all columns of D permuted so that the first s columns coincide with the s selected tests and appear in the order of selection. The rows of B are all these rows which share a common pattern e_{10}, $E_0, E_1, \ldots, E_{s-1}$ in ternary. The D_1 matrix of fig. 5(a) has been partitioned into its branch matrices at each decision level.

The iterative scheme developed for selecting the minimum number of additional tests necessary to distinguish all fault conditions of a given order is based upon the notion of weighting tests. The approach is consistent with the test selection algorithm developed by Chang[5] for optimizing binary decision trees. The criterion for test weighting depends upon the distribution of the "0's", "1's", and "2's" in each test pattern. Each test then partitions the set of fault conditions within each branch into three disjoint subsets. Therefore, any pair of these fault conditions constituted by taking one fault from one subset and one from either of the other two can be distinguished by the test. Since the number of pairs of fault conditions that can be selected from two subsets taking one from one subset and one from the other is the product of the number of elements in each subset and since with three subsets there are three distinct ways in which this can occur, the total number of pairs of branch fault conditions which a test can distinguish is given by:

$$w_j(e) = N_0 N_1 + N_1 N_2 + N_0 N_2 \qquad (8)$$

where N_0, N_1, and N_2 refer to the number of "0's", "1's", and "2's" respectively in branch matrix B_e's test pattern t_j for branch e. The branch matrix test pattern with the greatest weight distinguishes the largest number of pairs

16

of branch fault conditions. Theoretically a ternary decision tree will have 3^S branches at the s_{th} level. The sum of $w_j(e)$ over all branch matrices is called "the weight of test pattern t_j". This is denoted by W_j, where:

$$W_j = \sum_{e = 0}^{3^S-1} w_j(e) \qquad (9)$$

The following algorithm may be used to select the minimum number of additional tests for distinguishing all fault conditions of order k from the data matrix for this order.

1. Form the 3^S branch matrices of D_k, where s is the number of selected tests.

2. Compute W_j for all remaining tests.

3. Select a most weighted test. If its weight is greater than zero, return to step one. The test selected is t_{s+1}. If its weight is equal to zero , the process is completed for fault conditions of this order.

The process completes a given order in a finite number of iterations--in the worst case the number of fault conditions of this order or the number of unselected tests, whichever is smaller.

Employing the algorithm to the matrix D_1 of figure 5(a) indicates that tests T_1 and T_3 of figure 5(b) when coupled with T_0 are sufficient to diagnose all eight single defective module situations and are optimum because the minimum
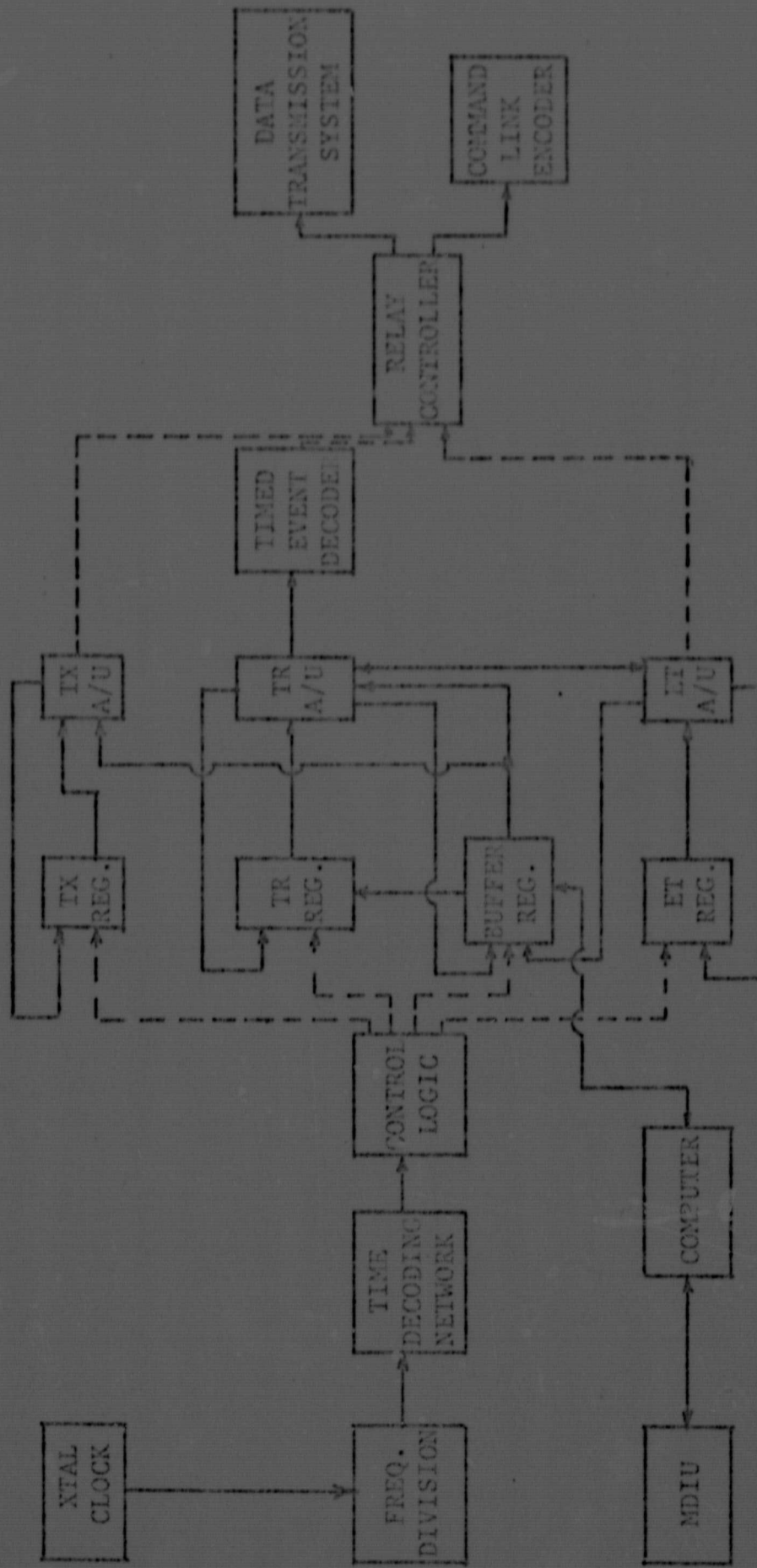
Figure 6

Functional Diagram of the Gemini Electronic Timer

number of binary decisions for eight objects, $\log_2 8$, is three. This three test sequence was carried to the D_2 matrix of fig 5(d). When the branch matrices were formed and a single iteration of the algorithm completed, it was found that test pattern t_3 associated with test T_4 [0 1 1 0] with a weight of six was sufficient to extend the test sequence to handle all correctable double fault conditions. Thus three reconfigurations of the system are suf-ficient to diagnose all eight single fault conditions and all twenty-four double fault conditions.

Simulated MABMR for the Gemini Electronic Timer

The functional diagram of the Gemini Electronic Timer appears in figure 6. The Manual Digital Indicator Unit (MDIU), Computer, Data Transmission System and Command Link Encoder are external systems interfacing with the Electronic Timer. Physically the electronic timer consists of seven interconnected mo-dules. One module contains the power supply while the other six consist en-tirely of logic circuitry. These latter six modules contain approximately three hundred gates and eighty memory elements. A Boolean model was developed for this circuitry and varified by computer simulation. Faults were inserted into this model to assist in locating the error detecting logic. With the development of models for the error detecting logic and intermodular switches, the Electronic Timer model was adapted to simulate BMR operation. Tests were developed for the six module subsystems using the algorithm of the previous section. The simulation of various fault conditions demonstrated the capa-bility of the MABMR system to diagnose detected faults to the module level, to reconfigure itself to by-pass the faulty modules, and to restore system operation.

SUMMARY AND CONCLUSION

A model assisted approach to bi-modular redundancy has been described for application to operational synchronous digital systems. The manner in which this approach provides continous error detection, diagnosis of all single fault conditions and all correctable multiple fault conditions, and self-repair, by automatic reconfiguration, have been detailed. The theoretical basis for the approach was presented and an algorithm was developed for generating an optimal sequence of diagnostic tests. The study concluded with a brief description of the simulation of a MABMR as applied to the Electronic Timer of the Gemini Time Reference System.

In concluding it is interesting to compare this approach to fault diagnosis with the conventional approach. Normally a sequence of inputs are applied to a system whose configuration remains static during diagnosis, as in the parallel redundant example. In the MABMR diagnostic procedure, a single pair of consecutive inputs are applied repeatedly, to a system which is undergoing a sequence of configurations. One significant advantage of this latter approach is that the testing sequence is largely independent of the circuitry within a module. Thus the same test sequence could be utilized effectively on two entirely different N-module BMR systems.

REFERENCES

[1] J. D. Brule, R. A. Johnson, and E. Kletsky, "Diagnosis of Equipment
Failures," IRE Trans. on Reliability and Quality Control, vol.
RQC-9, pp.23-24; April 1960.

[2] D. C. Roberts, "Increasing Reliability of Digital Computers,"
Computer Design Magazine, pp44-48; January 1969.

[3] D. B. Armstrong, "On Finding a Nearly Minimal Set of Fault Detection
Tests for Combinatorial Logic Nets," IEEE Trans. on Electronic
Computers, vol. EC-15, pp.66-73; February 1966.

[4] K. Maling and E. L. Allen, "A Computer Organization and Programming
System for Automated Maintenance," IEEE Trans. on Electronic
Computers, vol.EC-12, pp. 887-895; December 1963.

[5] H. Y. Chang, "An Algorithm for Selecting an Optimum Set of Diagnostic
Tests," IEEE Trans. on Electronic Computers, vol. EC-14, pp. 706-711;
October, 1965.