

NASA CASE ~~DA~~ 770-13086-1

PRINT FIG. ....

NOTICE

The invention disclosed in this document resulted from research in aeronautical and space activities performed under programs of the National Aeronautics and Space Administration. The invention is owned by NASA and is, therefore, available for licensing in accordance with the NASA Patent Licensing Regulation (14 Code of Federal Regulations 1245.2).

To encourage commercial utilization of NASA-owned inventions, it is NASA policy to grant licenses to commercial concerns. Although NASA encourages nonexclusive licensing to promote competition and achieve the widest possible utilization, NASA will consider the granting of a limited exclusive license, pursuant to the NASA Patent Licensing Regulations, when such a license will provide the necessary incentive to the licensee to achieve early practical application of the invention.

Address inquiries and all applications for license for this invention to NASA Patent Counsel, NASA Pasadena Office, Mail Code I, 4800 Oak Grove Drive, Pasadena, California, 91103. Approved NASA forms for application for nonexclusive or exclusive license are available from the above address.

AWARDS ABSTRACT

Inventor: Francis P. Mathur  
Contractor: Jet Propulsion Laboratory  
Contract NAS7-100

~~\_\_\_\_\_~~  
NASA Case # NPO-13086-1  
~~\_\_\_\_\_~~

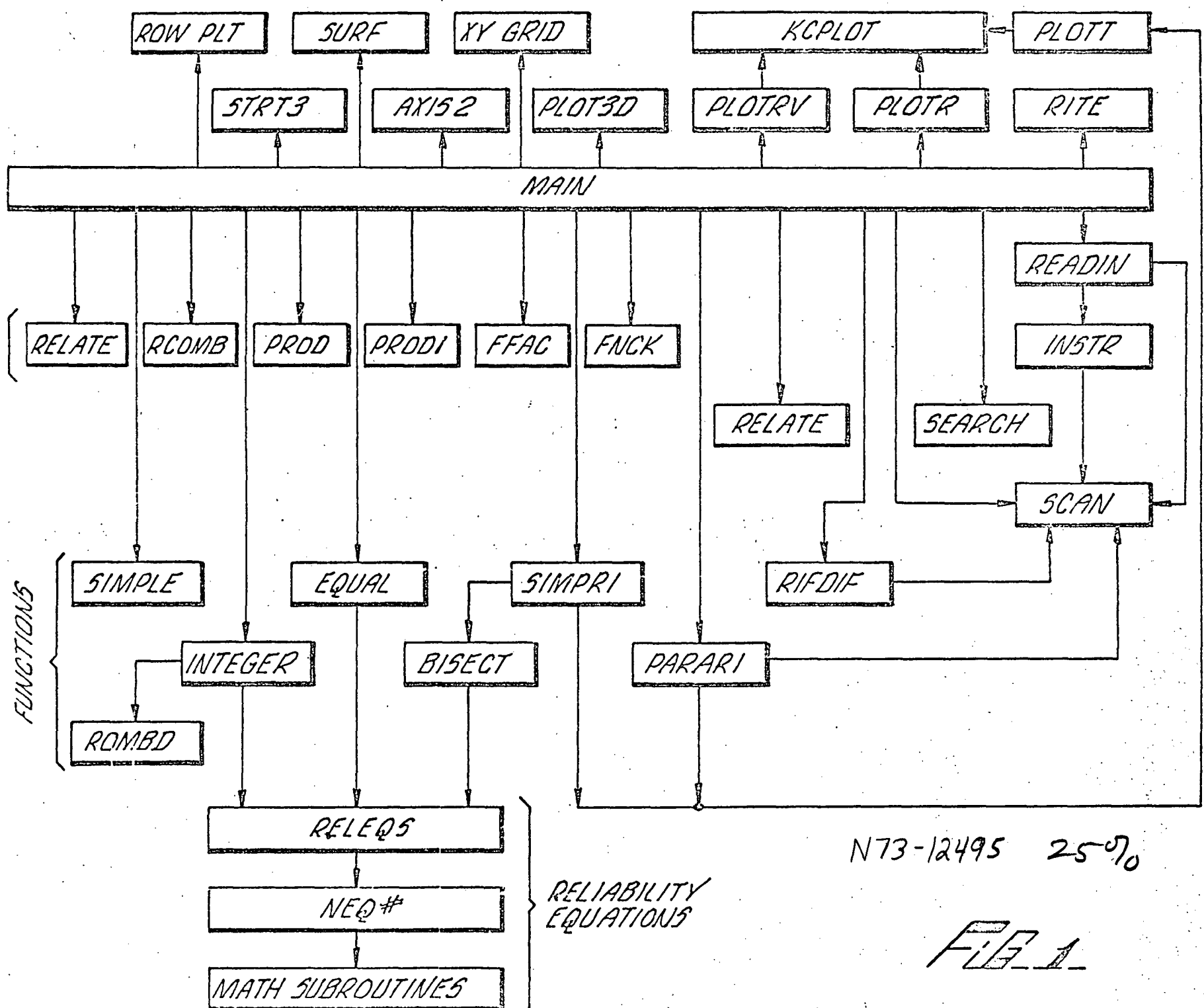
PROGRAM FOR COMPUTER-AIDED RELIABILITY ESTIMATION

This invention is directed to a computer program for estimating the reliability of self-repair and fault-tolerant systems with respect to selected system and mission parameters.

The computer program is capable of operation in an interactive "conversational" mode as well as in a batch mode and is characterized by maintenance of several general equations representative of basic redundancy schemes in an equation repository. Selected reliability functions applicable to any mathematical model formulated with the general equations, used singly or in combination with each other, are separately stored. One or more system and/or mission parameters may be designated as a variable. Data in the form of values for selected reliability functions is generated in a tabular or graphic format for each formulated model.

The novelty of the invention appears to lie in the provision of a computer program employing general equations that describe basic redundancy schemes and which may be readily used singly or in various selected combinations to formulate simple as well as complex models for evaluation. Further novelty is believed to rest in the use of separate repositories for the general equations and the reliability functions such that the equations are independent of the reliability functions and the equation repository is readily extended to include additional equations.

CALCULATIONS

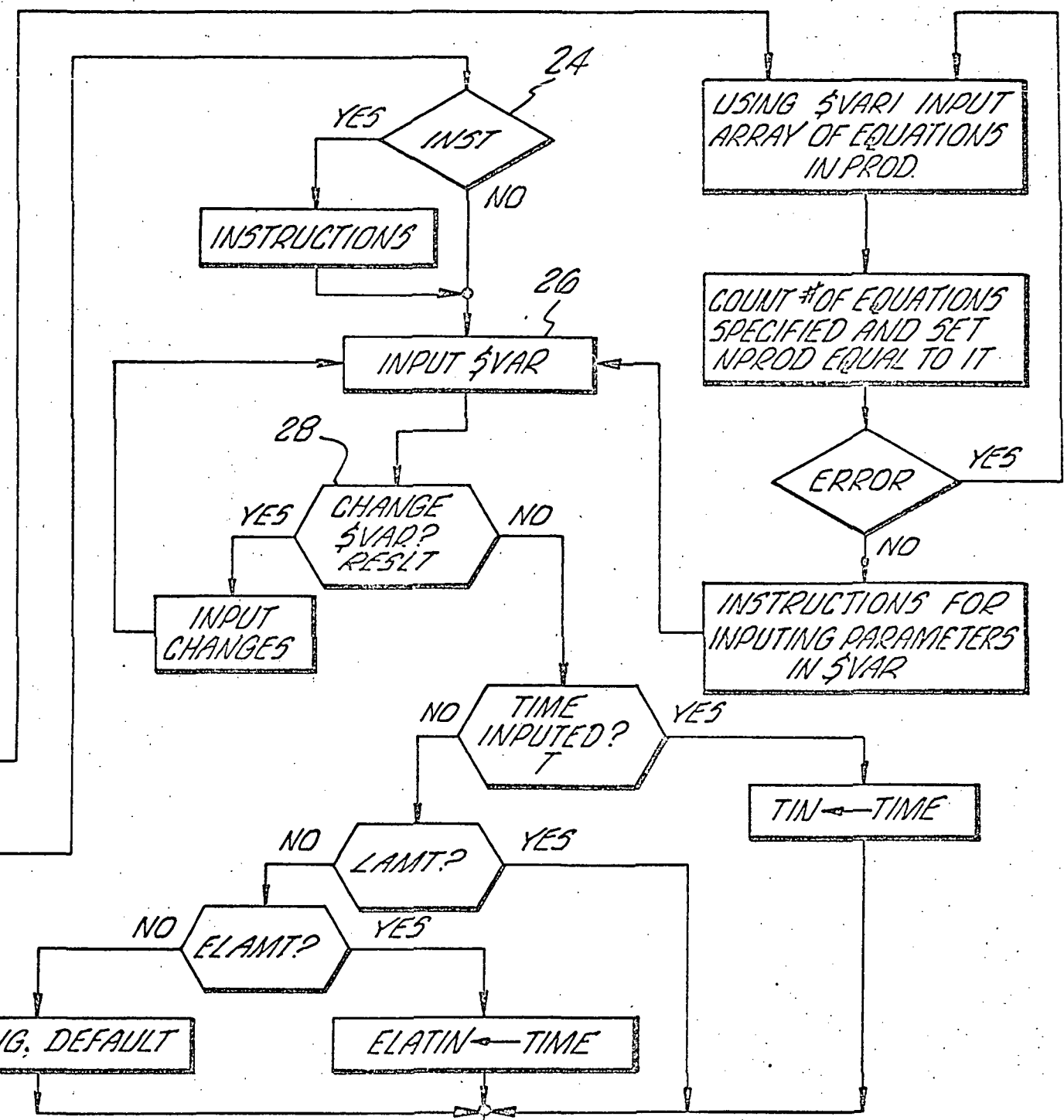
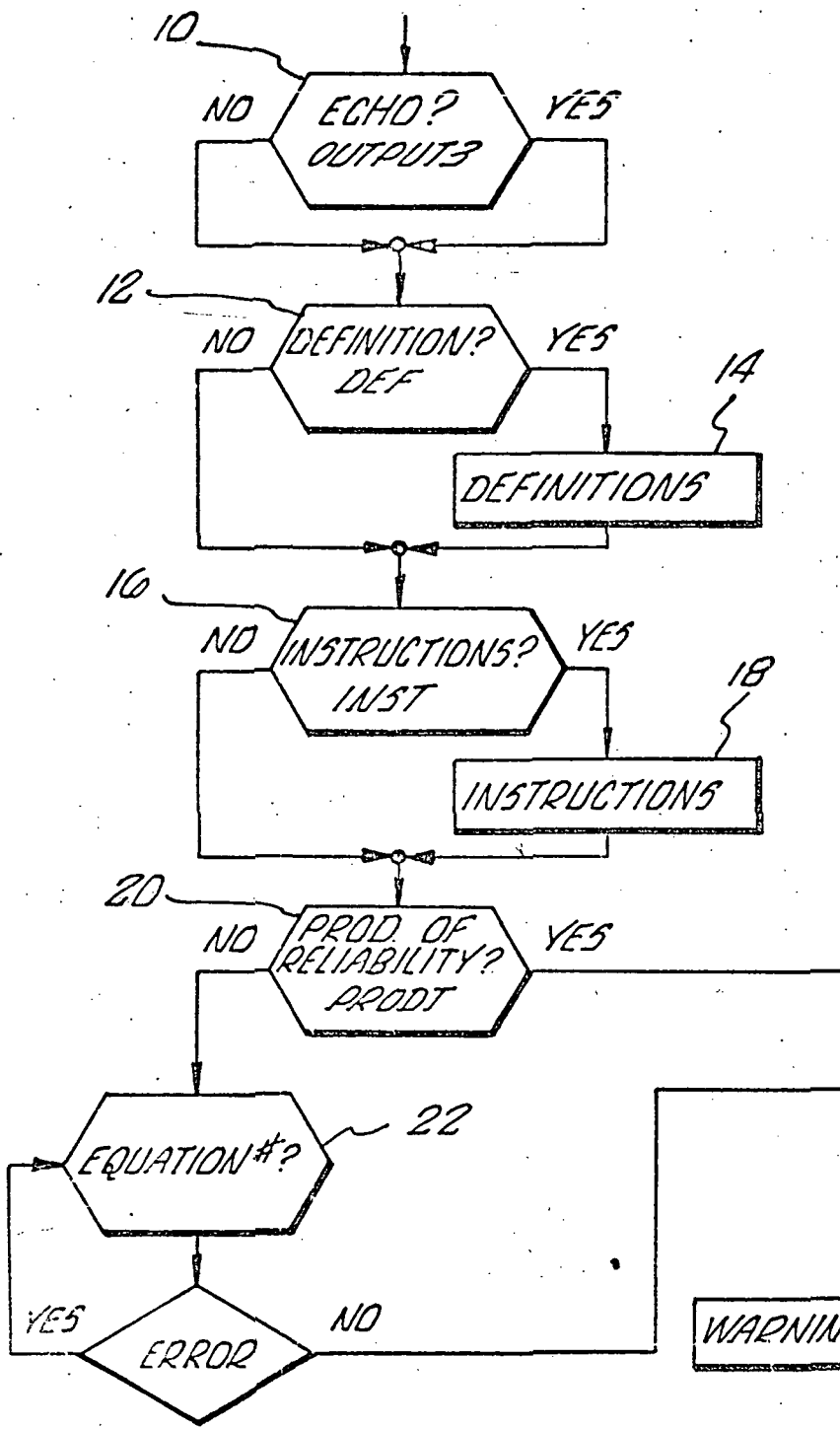


N73-12495 2570

RELIABILITY EQUATIONS

FIG 1

N73-12495



USING \$VARI INPUT  
ARRAY OF EQUATIONS  
IN PROD.

COUNT # OF EQUATIONS  
SPECIFIED AND SET  
NPROD EQUAL TO IT

ERROR

INSTRUCTIONS FOR  
INPUTTING PARAMETERS  
IN \$VAR

TIN ← TIME

ELATIN ← TIME

WARNING, DEFAULT

TO FIG. 3.

A

FIG 2

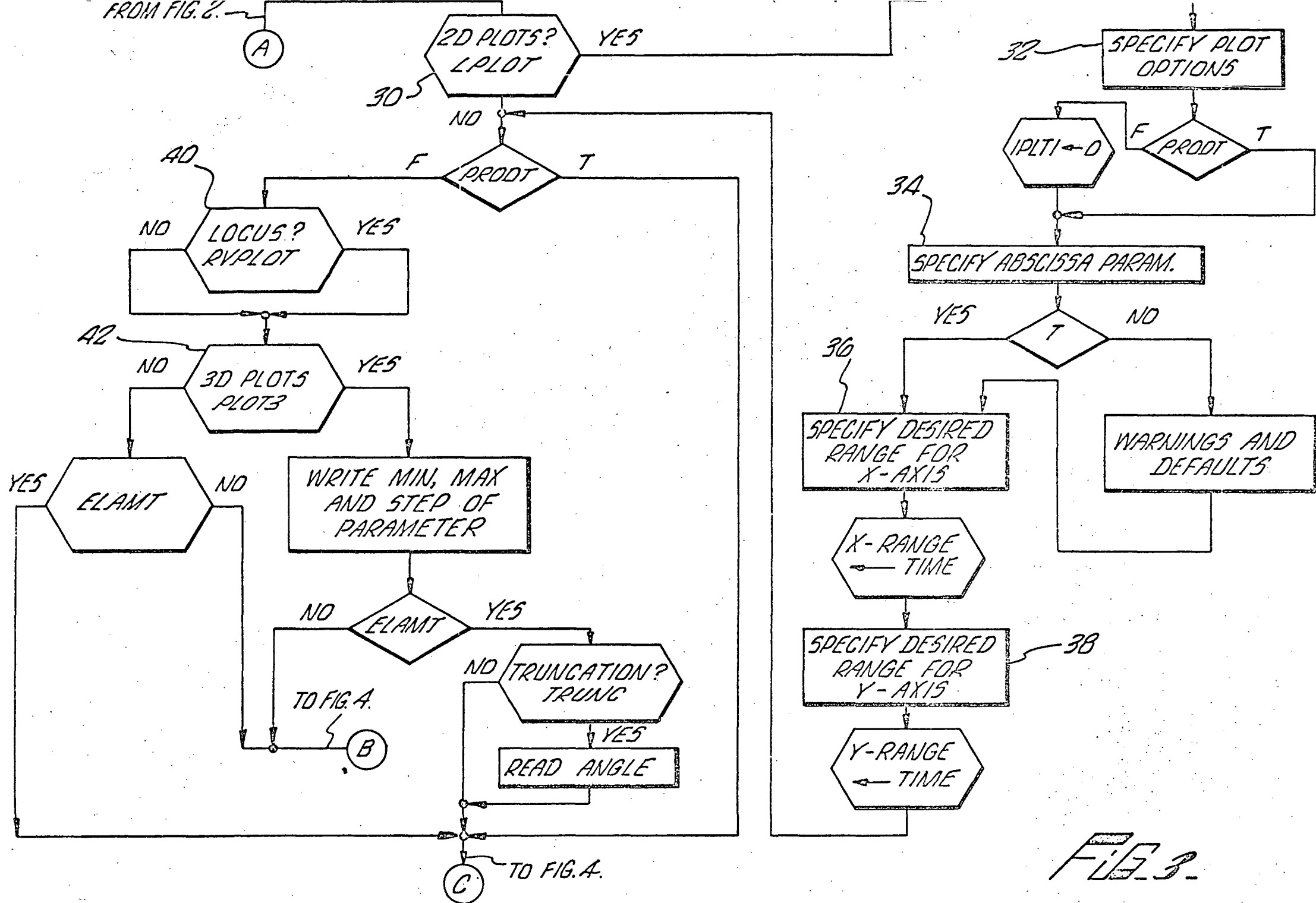


FIG. 3.

FROM FIG. 3.

(B)

FIG. 4.

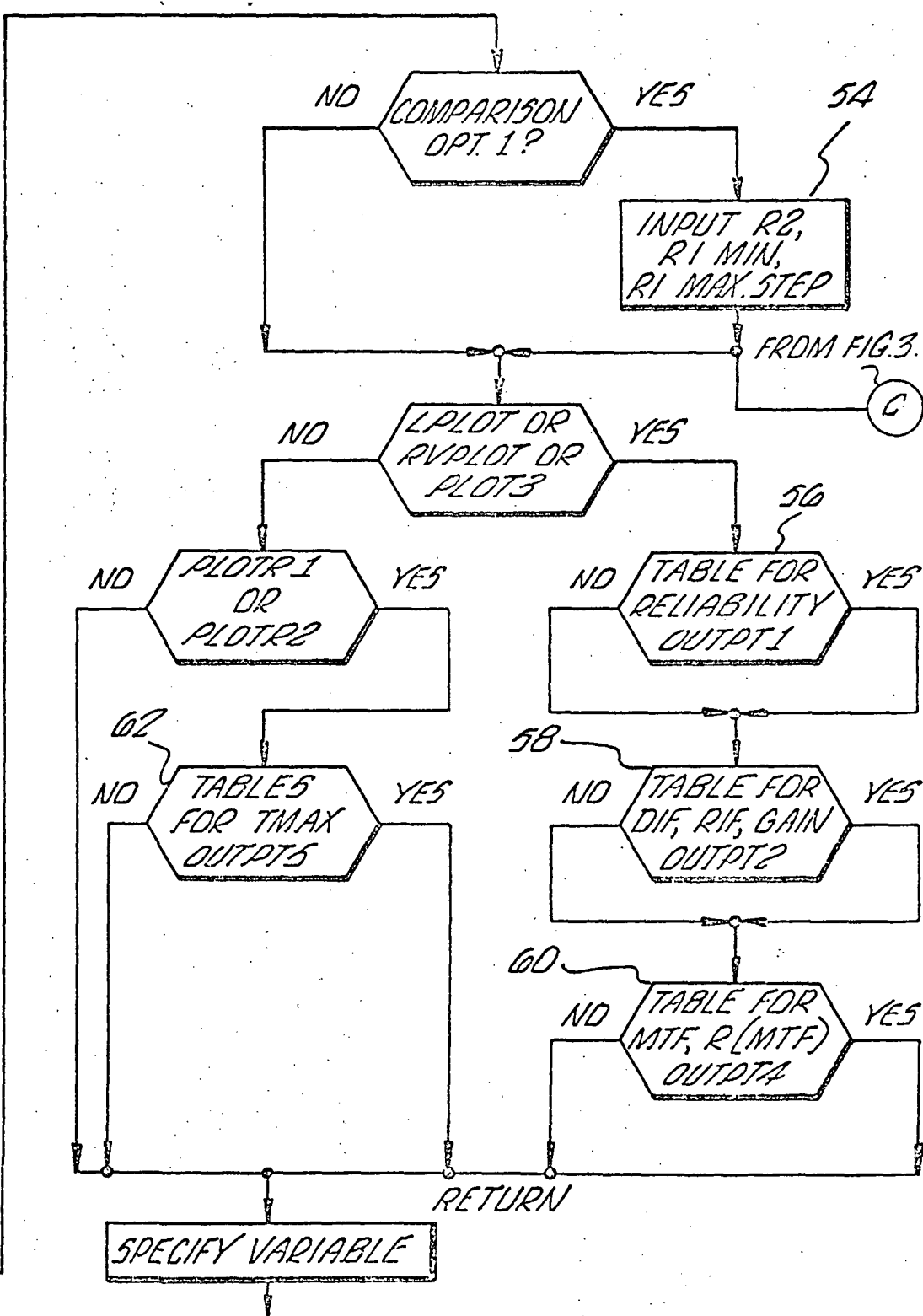
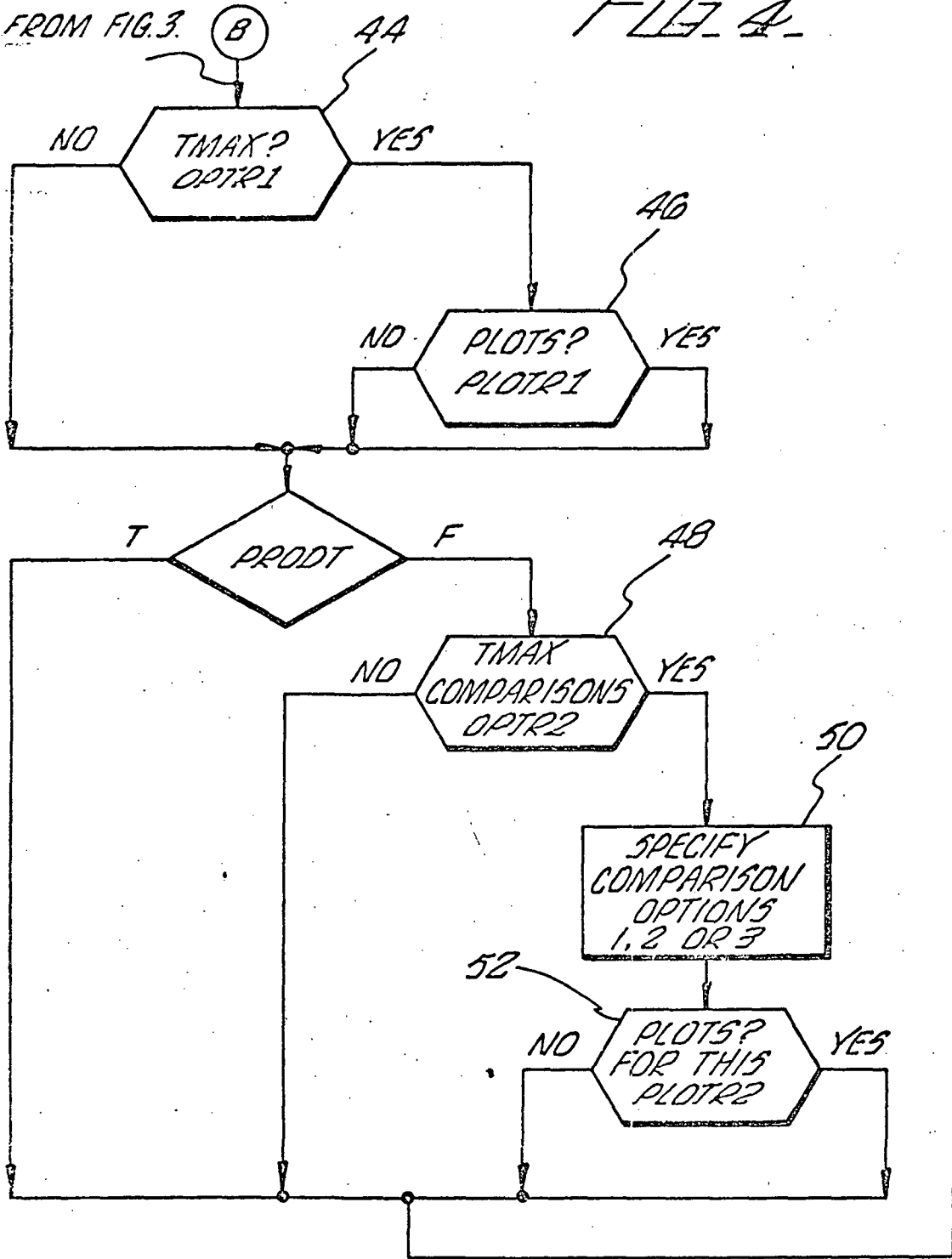
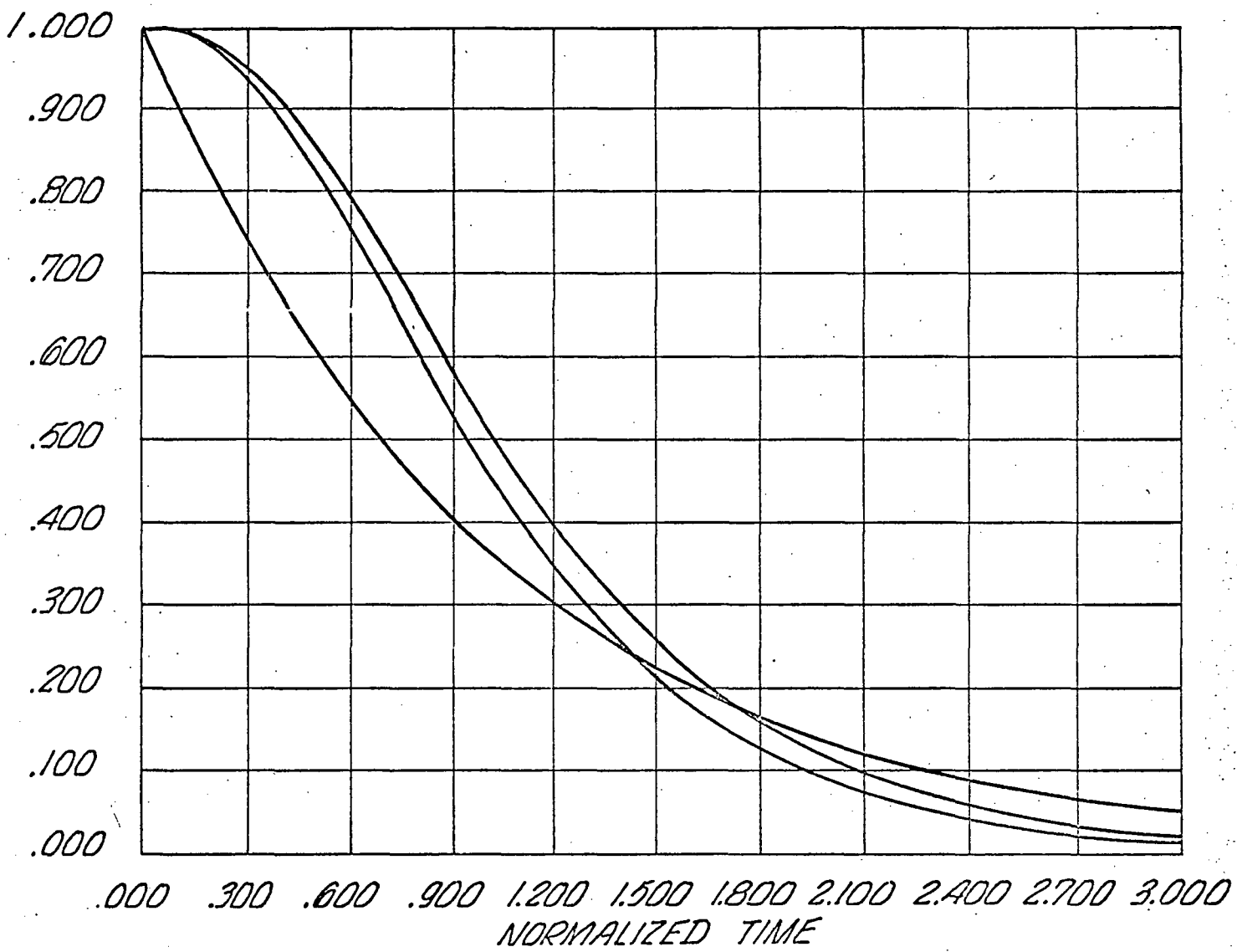


FIG. 5.

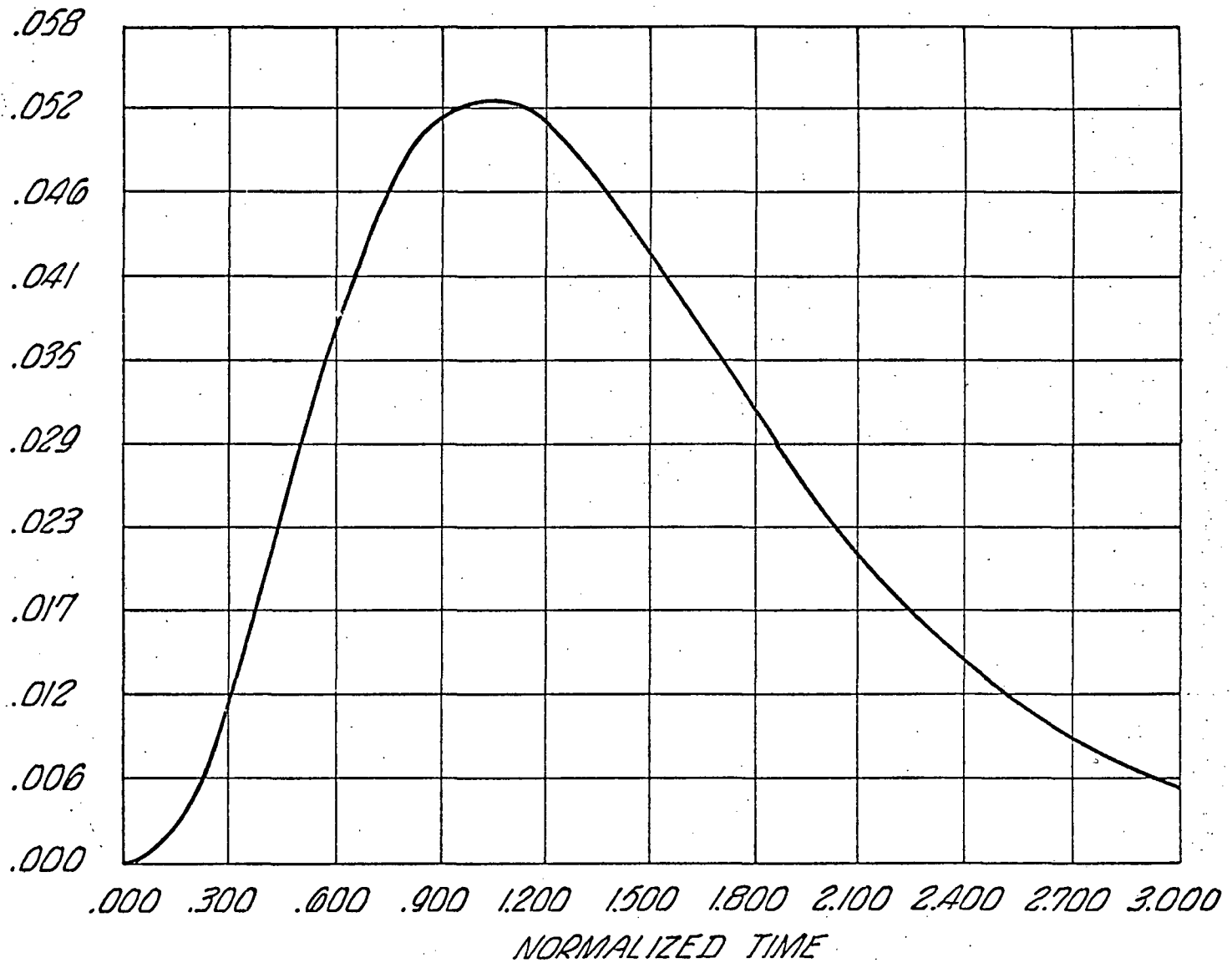


PLOT FOR EQUATION 1

FAMILY OF PARAMETERS IS  $K = .10000000+001 \quad .10000000+006$   
 (NI MEANS NOT INPUTED)

LAMBDA	MU	S	n	K	Q	C	RV	Z	W	P	MUT
NI	NI	1	1	NI	NI	NI	.10000000+01	1	1	.10000000+01	NI

FIG 6.



NORMALIZED TIME  
PLOT FOR EQUATION 1

FAMILY OF PARAMETERS IS  $K = .10000000+001 \quad .10000000+006$   
(NI MEANS NOT INPUTED)

LAMBDA	MU	S	T	K	Q	C	RV	Z	W	P	MUT
NI	NI	1	1	NI	NI	NI	.1000000+01	1	1	.1000000+01	NI



S P E C I F I C A T I O N

TO WHOM IT MAY CONCERN:

BE IT KNOWN THAT FRANCIS P. MATHUR, a citizen of the United States and residing in the County of Boone, State of Missouri, has invented a new and useful

PROGRAM FOR COMPUTER-AIDED RELIABILITY ESTIMATION of which the following is a specification

ABSTRACT OF THE DISCLOSURE

A computer program which effects computation of a plurality of reliability functions with respect to various system and mission parameters, is disclosed. The computer program is characterized by employing a separate equation repository and parameter storage which are independent of each other. Generalized equations are selectively used individually, or as complex products, to formulate mathematical models of self-repair or fault-tolerant systems to be evaluated with respect to selected system parameters. Reliability functions are able to be presented in a tabular and/or graphic format.

BACKGROUND OF THE INVENTION

1. Origin of the Invention

The invention described herein was made in the performance of work under a NASA contract and is subject to the provisions of Section 305 of the National Aeronautics and Space Act of 1958, Public Law 85-568 (72 Stat. 435; 43 U.S.C. 2457).

2. Field of the Invention

This invention is in the field of machine-performed processes. More specifically, the present invention concerns a computer program useful for simulating and evaluating self-repair and fault-tolerant organizations with respect to selected system

and mission parameters.

### 3. Description of the Prior Art

5 The design of ultrareliable fault-tolerant systems particularly suitable for long missions is required to satisfy the needs of spacecraft systems destined for outer space exploration. Such design of systems involving self-repair and fault-tolerance leads to the companion problem of quantifying and evaluating the survival probability of the system for the mission under consideration and under the constraints imposed upon the system.

10 Automated procedures that would enable the designer to rapidly model, simulate, and analyze preliminary designs and through man/machine symbiosis arrive at optimal and balanced fault-tolerant systems under the constraints of a prospective mission would greatly facilitate a system designer's job.

15 Several reliability evaluation programs are known in the prior art. Three of these programs are commonly known as the RCP, the RELAN and the REL70. The RCP is a reliability computation package developed by P. O. Chelson and has the capability of modeling a network of arbitrary series-parallel combinations of building blocks and analyzing the system reliability by means of probabilistic fault trees. A detailed description of the RCP program is found in "Reliability Math Modeling Using the Digital Computer", Jet Propulsion Laboratory, TR-32-1089, April 1967; and "Reliability Computation Using Fault Analysis", Jet Propulsion  
20 Laboratory, TR-32-1542, December 1971.

25 The RELAN is an interactive program which, like the RCP, models arbitrary series-parallel combinations; but in addition, allows a wide choice of failure distributions. RELAN has concise and easy to use input formats and provides elegant outputs such as plots and histograms. A detailed description of RELAN is  
30

provided by Computer Sciences Corporation publication entitled  
"RELAN: Reliability Analysis Package", CSC Sales Brochure  
No. 333, 1970.

5 The REL70 is also an interactive program but differs from  
the RCP and RELAN by being more adapted for evaluation of systems  
other than series-parallel configurations. For example, the  
REL70 is adapted to evaluate standby-replacement and triple  
modular redundancy systems. REL70 offers a large number of system  
parameters such as "coverage factor" (C) which is defined in  
10 the art as the probability of a system recovering from a failure  
given that the failure exists, and "quota" (Q) which is defined  
as the number of modules of the same type required to be operating  
concurrently. REL70 is primarily oriented towards the exponential  
distribution though it does provide limited capabilities for  
15 evaluating reliability with respect to selected mathematical  
distributions. The REL70 is slow in operation, however, speed  
compensation has been sought by incorporating the use of approx-  
imate versions of explicit reliability equations which are  
particularly applicable to short missions. A detailed description  
20 of the REL70 may be obtained by reference to "Design Techniques  
for Modular Architecture For Reliability Computer Systems" by  
W.C. Carter et al, IBM T.J. Watson Research Center Report  
No. 70-208-0002, March 1970; "Investigations in the design of an  
automatically repaired computer", by W.G. Bouricius et al, Digest  
25 of the First Annual IEEE Computer Conference, Sept. 1967, pp 64-67;  
and "Phase II of an architectural study for a self-repairing  
computer", by J.P. Roth et al, IBM Report SAMSO TR-67-106,  
Nov. 1967.

30 By comparison, the subject invention is a general program  
for evaluating fault-tolerant systems. The subject program is

general in that its reliability functions do not pertain to any one system or generalized equation representative thereof; but instead are applicable to all equations employed by the program to formulate specific mathematical models. Further, the reliability functions of the subject invention are applicable to any complex equations that may be formed by interrelating the basic generalized equations maintained for use by the program.

The use of an equation repository permits easy extension of the repository to include any other generalized equations that may be developed. Also, the use of "dummy" equations in the repository permits the timely insertion of any desired equation on a per case basis.

#### OBJECTS AND SUMMARY OF THE INVENTION

Accordingly, it is an object of the present invention to provide a computer program that may be used to quantify and evaluate the survival probability of self-repair and fault-tolerant systems with respect to selected system and mission parameters.

It is another object of the subject invention to provide a computer program which may be used to formulate mathematical models of selected self-repair and fault-tolerant system organizations.

It is a further object of the present invention to provide a computer program that permits computation of survival probabilities, mean life, and other selected reliability functions that are useful for predicting the reliability of selected model systems with respect to a prospective mission.

It is a yet further object of the present invention to provide an automatic procedure by which the reliability of selected self-repair and fault-tolerant systems can be quantitatively compared with competitive systems using a variety of

measures for comparison.

It is a still further object of the present invention to provide a computer program having the capacity to provide predictive reliability functions for models of fault-tolerant systems in tabular and/or graphic formats.

Briefly described, the present invention involves a computer program which may be used to compute reliability functions for hypothetical self-repair and fault-tolerant system organizations.

More particularly, the subject program is designed to provide computer-aided reliability estimation in the form of reliability functions for formulated mathematical models with respect to selected system and mission parameters. Generalized equations representative of basic systems are maintained in a repository which may be extended to include new equations on a temporary or permanent basis. Each mathematical model is formulated by using the generalized equations individually or in combination for complex systems. Values for selected reliability functions applicable to the formulated model are generated after entry of chosen system and mission parameters. Default values for certain common parameters are maintained for use in instances where a program user fails to specify a parameter value necessary to compute a requested reliability function. The resulting reliability functions may be automatically compared with other generated groups of reliability functions or with all other permutations of reliability functions that have been generated. Each group of reliability functions and all comparisons can be received in tabular or graphic form as desired.

The features that characterize the novelty of the present invention are set forth with particularity in the appended claims, both the organization and manner of operation of the

invention as well as the objects and many attendant advantages thereof may be best understood by reference to the following detailed description considered in conjunction with the accompanying drawings.

5

#### BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a schematic block diagram that generally illustrates a structural implementation of a computer program in accordance with the present invention.

10

Figures 2, 3 and 4 form a flow chart in block diagram form which illustrates the manner in which reliability functions, and tables and graphs thereof, are generated by a computer program in accordance with the present invention.

Figures 5 and 6 are exemplary plots of selected output data that can be generated in accordance with the present invention.

15

#### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

##### Functional Description

20

A computer program in accordance with the present invention serves as a computer-aided reliability design tool to designers of ultra-reliable fault-tolerant systems by facilitating reliability computation, data generation, and comparative evaluation. The results provided by the program are available as tabular printouts, graphical two dimensional plots, and graphical three dimensional projections.

25

Essentially, the program involves a repository of mathematical equations that define the basic redundancy schemes that are used to provide fault-tolerant systems. These equations under program control are interrelated to generate desired mathematical models to fit the architecture of a fault-tolerant system under evaluation. The mathematical model is then supplied with chosen system and/or mission parameter values with certain

30

parameters being used as variables. The model may then be evaluated to yield values for a specified independent parameter or for selected reliability functions.

5 The program has three basic modes of operation. These modes may be referred to as the "conversational", or interactive mode, the batch mode, and the remote-started batch mode. In the "conversational" mode, the program may be interactively accessed by users from remote teletype facilities or other communications consoles to perform reliability analysis in  
10 "real time". Required inputs are in the form of a selection of one or more reliability equations followed by queries and answers on the various parameters of interest and their behavior with respect to mission time, normalized time, non-redundant system reliability, failure rates, inverse dormancy factors,  
15 fault-coverage, cascades of units, and allocated spares.

In the batch mode, the evaluation is intended to be conducted after the equation selection and system parameters are submitted off-line. In this mode, no dynamic changes to the user requirements can be made. The primary benefit of the  
20 batch mode is expeditiousness and it is intended for users who know exactly what is wanted and hence need not spend time sitting at a console to input his queries.

The remote-started batch mode is similar to the batch mode except that, instead of submitting the job as a deck of  
25 punched cards, the deck entry can be made via a console.

The reliability of any fault-tolerant system may be quantitatively evaluated, described, and compared in terms of various reliability functions. The reliability functions that the subject program employs, or can employ, with respect to  
30 selected equations and parameters are provided by Table I

hereinbelow:

Table I

	<u>Program Word</u>	<u>Reliability Function</u>
	REL	system reliability
5	UNREL	system unreliability = $(1 - \text{REL})$
	SIMREL	non-redundant simplex reliability = $\text{ELAMT}$
	SIMGAIN	gain in reliability with reference to a simplex system $\text{REL}/\text{SIMREL}$
10	SIMRIF	reliability improvement factor with reference to a simplex system $(1 - \text{SIMREL})/(1 - \text{REL})$
	DIFF	difference in reliabilities $R(\text{system}2) - R(\text{system}1)$
15	RIF	reliability improvement factor $[1 - R(\text{system } 1)]/[1 - R(\text{system } 2)]$
	GAIN	gain in reliability $R(\text{system } 2)/R(\text{system } 1)$
	SIMTMAX	maximum mission length of a simplex system for a given mission reliability R
20	TMAX	maximum mission time length of the system for a given mission reliability R
25	SIMTIF	time improvement factor with reference to the simplex system $\text{TMAX}/\text{SIMTMAX}$
	RATIF	time improvement factor $\text{TMAX}(\text{system } 2)/\text{TMAX}(\text{system } 1)$
30		



TABLE 1 Con't.

<u>Program Word</u>	<u>Reliability Function</u>
MTF	mean life
R(MTF)	reliability at the mean life.

5 Besides providing the reliability functions listed in Table I, the program can also perform an evaluation of complex reliability systems formed by cascading basic systems by placing multiple basic systems in series, by jointly cascading and placing in series multiple basic systems, or by taking the products of basic reliability equations. Further the program can be made to provide a locus of values of reliability of a restoring organ (RV) such that the system reliability equals the unit reliability.

10 Table II hereinbelow is a tabular presentation of program words for certain common parameters which are provided for by the subject program.

TABLE II

<u>Program Word</u>	<u>Parameter/Descriptions</u>
T	mission time
R	system reliability
20 S	the total number of spares
n	$(N - 1)/2$ where N is the total number of multiplexed units
K	inverse dormancy factor (= $LAMBDA/MU$ )
25 C	coverage factor, which is the conditional probability of a system recovering given a failure occurrence
30 Q	quota, the number of identical units in a simplex system

TABLE II 'Con't.

<u>Program Word</u>	<u>Parameter/Descriptions</u>
W	number of cascaded units
Z	number of identical systems in series
P	probability of a unit failing to zero
RV	reliability of the restoring organ
MU	unpowered failure rate of a simplex system (=K/LAMBDA)
LAMBDA	powered failure rate of a simplex system (K x MU)
LAMT	maximum normalized mission time
ELAMT	exp(- LAMT)

The outputs generated by the subject program are in the form of tables and/or plots which may be optionally selected by the user. The plotting may be actually performed off-line on any suitable plotter available in the prior art. For example, a Stromberg Carlson 4020 plotter has been found to be suitable for this purpose. Two or three dimensional plots are available of which the X and Y axes may be constrained to desired values to limit the plotting region. The truncation of three dimensional plots with plane surfaces is also possible in accordance with the subject invention. Most of the computer data is presented in a tabular format. The five available tabular formats (as a function of the selected system parameter) are listed in Table III hereinbelow.

TABLE III

<u>Format</u>	<u>Data in Tabular Format</u>
1.	<u>T</u> or <u>LAMT REL UNREL SIMREL SIMGAIN SIMRIF</u>

TABLE III Con't.

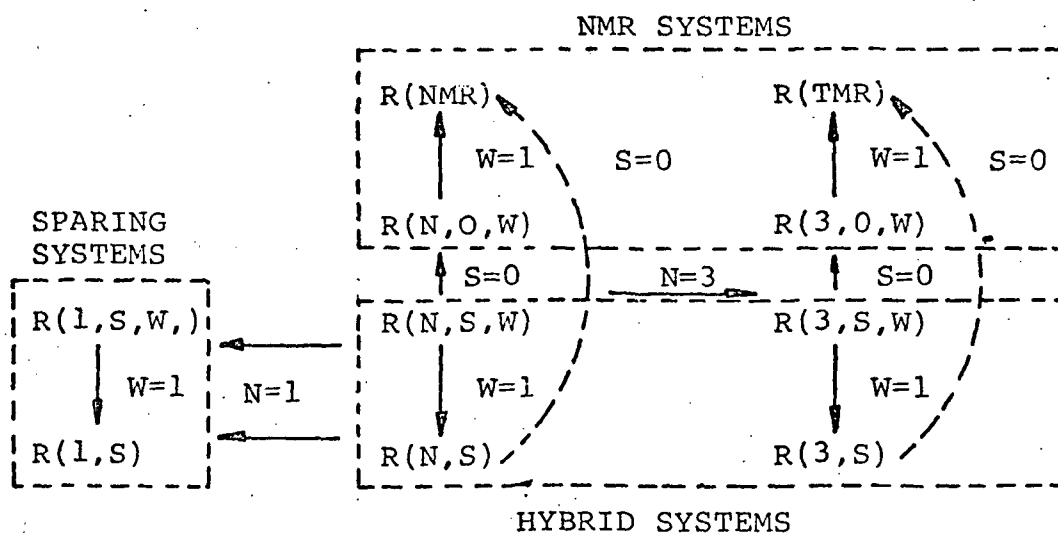
<u>Format</u>	<u>Data in Tabular Format</u>
2.	<u>T</u> or <u>LAMT</u> <u>DIFF</u> <u>RIF</u> <u>GAIN</u>
3.	<u>ELAMT</u> vs. <u>RV</u> (for <u>SIMGAIN</u> = 1)
4.	<u>R1</u> <u>SIMTMAX</u> <u>TMAX</u> <u>SIMTIFF</u> (for some R2)
5.	<u>R1</u> <u>TMAX1</u> <u>TMAX2</u> <u>RATIF</u> (for some R2)

Data presented in tabular format is accompanied by the values of mean life and reliability at the mean life which are printed out immediately following the reliability calculation.

Notation for System Configurations

A unifying notation was developed by the inventor to write the equations for the various systems configurations using selective massive, or hybrid redundancy. A detailed discussion of such unifying notation is provided in a publication entitled "Reliability Modeling, Analysis and Prediction of Ultra-Reliable Fault-Tolerant Digital Systems" by F. P. Mathur, Digest of the 1971 International Symposium on Fault-Tolerant Computing, Pages 79-82, March 1-3, 1971.

Briefly, however, the diagram hereinbelow generally illustrates the interrelationships between the notations for "sparing" systems, "NMR" systems and "hybrid redundant" systems.



Referring to the above diagram, a hybrid redundant system  $H(N,S,W)$  is said to have a reliability  $R(N,S,W)$ . If the number of spares in the system is zero, i.e.,  $S=0$ , then the hybrid system is reduced to a cascaded NMR system whose reliability expression is denoted by  $R(N,O,W)$ . In a case where there are no cascades, a so-called cascaded NMR system is further reduced to  $R(N,O,1)$  or more simply to  $R(NMR)$ . The term  $W$  may be elided if  $W = 1$ . The sparing system has a reliability of  $R(1,S)$  and essentially consists of a single basic unit with  $S$  spares.

As noted in the referenced Digest of the 1971 International Symposium, a notational convention may be used wherein an asterisk over the "R" indicates that the unreliability of a restoring organ has been taken into consideration, i.e.,  $R^*(NMR)$ . If the asterisk is elided, it is to be assumed that the restoring organ has an infinitesimal probability of failure. This notational convention is primarily applicable to those systems that require restoring organs for their implementation.

#### Reliability Equations

As earlier mentioned, the subject program employs an equation repository that serves to maintain the basic system equations separate from other functions and/or parameters. In this manner the equations that are used to formulate the mathematical models of fault-tolerant systems to be evaluated are entirely independent of the reliability functions which are used to describe the mathematical models with respect to selected system and mission parameters.

The subject program has been defined to have a capacity for seven different basic equations to represent the basic redundancy schemes. Of course a greater capacity would have been possible. More complex equations representing more complex

systems may be formed by combining the basic equations. Of the seven equation capacity, five equations have been implemented and the allotted spaces for the remaining two equations have been preserved to permit future extensions. The five equations maintained in the repository and the basic fault-tolerant system organizations corresponding thereto are as follows:

(1) Equation 1 is the general reliability equation for hybrid redundant systems. Standby-replacement systems using selective or dynamic redundancy in combination with the general TMR systems result in the class of redundant systems designated as being hybrid redundant. Typical hybrid redundant systems would include NMR(N,0) systems and TMR(3,0) systems plus cascaded or partitioned versions, and series strings of the same. A detailed analysis and discussion of such hybrid redundant systems may be obtained by reference to the following articles, "Reliability Modeling and Analysis of a Dynamic TMR System Utilizing Standby Spares", by F.P. Mathur, Proc. of the 7th Annual Allerton Conference on Circuit and System Theory, University of Illinois, Urbana, Pages 243-252, October 8-10, 1969; and "Reliability Analysis and Architecture of a Hybrid Redundant Digital System: Generalized Triple Modular Redundancy with Self-Repair", by F.P. Mathur, et al, AFIPS Conference Proceedings (Spring Joint Computer Conference), Volume 36, Atlantic City, New Jersey, May 5-7, 1970.

The Hybrid (N,S) system consists of a NMR core with an associated bank of S spare units such that when one of the N active units fails, the spare unit replaces it and restores the NMR core to the all-perfect state. The physical realization of such a system is arrived at by means of disagreement detectors which compare the system output from the restoring organ with the

outputs of each one of the N active units. Upon the detection of  
 a disagreement a signal is transmitted to a switching net which  
 replaces the unit that disagreed by switching it out and switching-  
 in one of the spares. Should the spare unit have failed in the  
 dormant mode, upon being switched-in the disagreement would still  
 exist and the switching net would switch-in one of the remaining  
 spares. The hybrid (N,S) system reduces to a single NMR system  
 when all the spares have been exhausted. Notationally, a hybrid  
 (3,0) system is equivalent to a TMR system. Thus, from the  
 standpoint of mathematical modeling, the classical NMR systems  
 form a proper subset of the hybrid-redundant systems. The equation  
 representing the above indicated family of hybrid-redundant systems  
 is as follows:

$$\begin{aligned}
 R^*(N,S) &= \left[ R_s^{N/W} R_s^{1/W} \left[ 1 + \sum_{j=0}^{S-2} \binom{NK+S}{j+1} \left( \frac{1}{R_s^{1/W}} - 1 \right)^{j+1} + \sum_{i=0}^n \binom{N}{i} \left( \frac{NK+S}{S} \right) \sum_{l=0}^1 \right. \right. \\
 &\quad \left. \left. \times \frac{\binom{i}{l} (-1)^{i-l}}{\binom{Kl+S}{S}} \left[ \left( \frac{1}{R_s^{S/W} R_s^{l/W}} - 1 \right) - \sum_{j=0}^{S-2} \binom{Kl+S}{j+1} \left( \frac{1}{R_s^{1/W}} - 1 \right)^{j+1} \right]_{RV} \right]^{WZ} \right. \\
 &\quad \left. \text{for } 1 \leq K < \infty \text{ and } S = 0 \right. \\
 &= \left. \left[ R_s^{N/W} R_s^{1/W} \left[ 1 + (NK+1) \sum_{i=0}^n \binom{N}{i} \sum_{l=0}^j \binom{i}{l} \frac{(-1)^{i-l}}{(Kl+1)} \left( \frac{1}{R_s^{1/W} R_s^{l/W}} - 1 \right) \right]_{RV} \right]^{WZ} \right. \\
 &\quad \left. \text{for } 1 \leq K < \infty \text{ and } S = 1 \right.
 \end{aligned}$$

The related equations corresponding to the case  $K = \infty$  may  
 be found by reference to "Reliability Modeling and Architecture  
 of Ultra-Reliable Fault-Tolerant Digital Computers", by F. P. Mathur,  
 Ph.D. Thesis, University of California, Los Angeles, Computer

Sciences Dept., June 1970.

(2) Equation 2 is the general reliability equation for standby-replacement systems. Included in this category would be "K-out-of-N" systems described in detail in "Phase II of an Architectural Study For a Self-Repairing Computer", IBM report SAMSO TR-67-106, November 1967, Simplex systems, and Series string and cascaded versions of the same. A detailed description of standby-replacement systems may also be obtained by reference to the above-mentioned Ph.D. Thesis entitled "Reliability Modeling and Architecture of Ultra-Reliable Fault-Tolerant Digital Computers" written by the inventor of the present invention. The equation representing the standby-replacement systems is as follows:

$$R(l,S) = \left[ R^{Q/W} \left\{ 1 + \sum_{i=1}^S \left[ \frac{c^i}{i!} \left( 1 - R_s^{1/W} \right)^i \prod_{j=0}^{i-1} (QK + j) \right] \right\} \right]^{WZ}$$

for  $1 \leq K < \infty$

$$= \left[ R^{Q/W} \sum_{i=0}^S \frac{(CQ\lambda T/W)^i}{i!} \right]^{WZ}$$

for  $K = \infty$

(3) Equation 3 has been left blank to permit insertion of a new equation.

(4) Equation 4 is the general reliability equation for Hybrid/Simplex redundant systems. Included in this category are TMR/Simplex systems as well as series string and cascaded versions of the same. A detailed description of Hybrid/Simplex systems is available by reference to "Reliability Modeling, Analysis and Prediction of Ultra-Reliable Fault-Tolerant Digital Systems",

by F.P. Mathur, Digest of the 1971 International Symposium on Fault-Tolerant Computing, pages 79-82, Pasadena, California, March 1-3, 1971. Generally, the hybrid-redundant system H(3,S) uses the conventional TMR system along with a bank of standby spares. A variant of the TMR scheme called the TMR/Simplex system yields increased reliability by adopting the strategy of a triplicated majority voted system where upon the first failure of a unit, that unit is discarded, and one of the two remaining good units is substituted while the other is also discarded. The system is then operated in a simplex mode. Now if a hybrid-redundant scheme is devised which combines standby-replacement units with the above variant of a TMR system -- in the same manner as was done for the H(N,S) system described previously -- a new scheme called Hybrid/Simplex redundancy results. The equation representing such Hybrid/Simplex systems is as follows:

$$R(3,S)_{sim} [T] = R^3 R_s^S \left\{ 1 + 1 \cdot 5 \left( \frac{1}{R^2 R_s^S} - 1 \right) \prod_{i=1}^S \left( \frac{3K + i}{2K + i} \right) \right.$$

$$\left. - \prod_{j=1}^S \frac{(3K + j)}{j} \sum_{i=0}^{S-1} \binom{S}{i} (-1)^i \left( \frac{1}{R^{S-1} R_s} - 1 \right) \frac{3K^2}{(2K + i)(3K + i)} \right\}$$

for  $S > 0$  and  $\mu > 0$

$$\text{and } = (1 \cdot 5)^{S+1} R - R^3 \sum_{i=1}^S \frac{(3\lambda T)^{S+1-i}}{(S-i)!} \left[ (1 \cdot 5)^i - 1 \right] - R^3 \left[ (1 \cdot 5)^{S+1} - 1 \right]$$

for  $S > 0$  and  $\mu = 0$

also, using our notational convention:

$$R^*(3,S)_{sim} = R_v \cdot R(3,S)_{sim}$$



(5) Equation 5 is the general reliability equation for TMR systems where the probability of a unit failing to logical one or logical zero is parameterized. Series string and cascaded versions of such TMR systems are also represented by Equation 5. The above referenced article entitled "Reliability Modeling and Architecture of Ultra-Reliability Fault-Tolerant Digital Computers" also describes in detail such TMR systems. Equation 5 for such TMR systems is as follows:

$$R^*(3,0) = \left\{ RV \left[ 3R^{2/W} - 2R^{3/W} + 6P(1-P) R^{1/W} (1 - R^{1/W})^2 \right] \right\}^{WZ}$$

(6) Equation 6 is the general reliability equation for Simplex systems and is as follows:

$$R(1,0) = \left[ R^{1/W} \right]^{WZ}$$

(7) Equation 7 has been left blank to permit insertion of a new equation.

The aforementioned five of seven have been included in the equation repository of the subject invention. Equations three and seven are the earlier characterized "dummy" equations and may be placed in any of the seven positions.

The total number of equations has been restricted to seven. The equations are intended to provide the most general mathematical expressions for the corresponding basic systems which can be used to parameterize mission time, failure rates, dormancy factors, coverage, number of spares, number of multiplexed units, number of cascaded units, and number of identical systems in series. Complex systems are modeled by taking any of the above equations in series with another.

#### Reliability Theoretic Functions

The reliability equations in the repository may be evaluated as a function of absolute mission time (T), normalized mission time (LAMBOA x T), system reliability (R) or any other system

parameter that may be applicable. The set of reliability functions defined in the program are applicable to any of the equations in the repository, taken singly or in combination. This independence of the equations from the reliability functions to be applied to the equations imparts a significant degree of generality to the program. For example, the equation repository may be upgraded without affecting the repertoire of functions.

The various reliability functions useful in the evaluation of fault-tolerant computing systems is presented in detail in an article written by the inventor entitled "On Reliability Modeling and Analysis of the Ultra-Reliable Fault-Tolerant Digital Computers", Special Issue on Fault-Tolerant Computing, IEEE Transaction on Computers, Volume C-20, No. 11, November 1971, pages 1376-1382. In the article, the measures of reliability are defined, characterized into the domains of probabilistic measures and time measures, and their effectiveness compared. As tabulated in Table I, hereinabove, among the various measures of reliability that the user may request for computation are the system mean life (MTF), the reliability at the mean life  $R(MTF)$ , the gain in reliability over a simplex system or some other competitive system (GAIN) and the reliability improvement factor (RIF).

#### Operational Features

Although the subject program is primarily an interactive program, i.e., "conversational mode", it may be run in the batch mode if the user prespecifies that protocol explicitly. In the interactive mode the program is designed to assume minimum knowledge on the user's part.

Default values are provided for many of the parameters that a user should normally supply. This feature safeguards the user and also makes usage of the program simpler since the logical

default values are available for conventionally used parameters should a user fail to input required values.

The following parameters if not inputted when required by the subject program are assigned the following default values as follows:

S = 1, N = 1, B = 1, W = 1, Q = .1000, K = 1.000, C = 0.000...000  
STEP = 1.000, ELAMT = 1.000, P = 1.000, MIN = 0.000,  
RV = 1.000.

Instructions are provided by the program as an option to permit an experienced user to circumvent the instructions to operate in a fast mode. Also definitions of reliability terms and abbreviations used in the program may be optionally requested. Finally, an optional "echo" feature that echoes a user's responses back to the terminal is provided.

#### Operational Limitations

Certain constraints have been designed into the program to satisfy practical limitations. Specifically, in formulating complex models, a maximum of ten equations can be involved in accordance with the present invention. The maximum number of iterations of any parameter values is 16. The array dimension of any parameter is 121 which means that if, for example, the mission time parameter T is desired to be incremented from a minimal value 0 to 12, then the minimum allowable increment in step size would be 0.1. Finally, with respect to the inverse dormancy factor (K), any value above and including 100,000 is equivalent to setting K = infinity. These constraints as a practical matter may be changed and are primarily imposed by memory storage requirements. For example, the maximum memory capacity available without having to resort to segmentation is 65,000 words.

### Parameter Handling

The system parameters, LAMBDA, Mu, S, N, K, Q, C, RV, Z, W and P are two dimensional parameter arrays, dimensioned as being 16 x NPT (short for "number of products"). As earlier mentioned, sixteen is the maximum number of values that any one parameter may be assigned in the VARIABLE namelist notation. The NPT pertains to the total number of equations that may be used in forming the product. If a complex equation is not being formed, then  $NPT = 1$ . Also as earlier mentioned, the maximum value that NPT can currently take is 10. Thus the rows of the parameter matrices may contain the values of the parameter while the columns may contain the index of the equation numbers (with reference to the order in which they were entered) that these parameters pertain to.

The time pertinent parameters, such as Time, LAMT, and ELAMT are single valued. Their values are the maximum values that the parameter is to take. The incremental steps at which computations are to be performed are specified by assigning a value to the variable STEP.

### Model Formulation-Example 1

A typical problem submitted for program analysis may be as follows: Given a simplex system with 8 equal modules which is made fault-tolerant by providing two standby spares for each module, where each module has a constant failure rate of 0.5 failures per year, the spares have a dormancy factor of 10, and the applicable coverage factor is 0.99, evaluate the system survival probability in steps of 1/10th of a year for a maximum mission duration of 12 years. It is required that the system reliability be compared against the simplex or non-redundant system and that all these results be tabulated and also plotted.

It is further required that the mean life of the system, as well as the reliability at the mean life, be computed. It is of interest to know the maximum mission duration that is possible while sustaining some fixed reliability objective and to display the sensitivity of this mission duration with respect to variations in the tolerable mission reliability.

It is also required that the above analysis be carried out for the case where three standby spares are provided and these configurations of three and two spares be compared and the various comparative measures of reliability be evaluated and displayed.

The above problem formulation is entered into the program by stating that Equation 2 (which models standby-replacement systems) is required. The pertinent data ( $S = 2, 3$ ;  $Z = 8$ ;  $K = 10$ ;  $T = 12.0$ ;  $LAMBDA = 0.5$ ;  $C = 0.99$ ;  $STEP = 0.1$ ) is inserted into the program between the VARIABLE namelist delimiters \$VAR...\$END.

The above example illustrates the complexity of problems that may be posed to the program, and the simplicity with which the specifications are entered. The reliability functions to be performed on the above specified system may be acknowledged interactively by answering YES or NO, on the demand terminal, to questions posed by the program from time to time.

#### Model Formulation-Example 2

Another example would be: given a standby-replacement system with one spare ( $S = 1$ ) and a maximum normalized mission time of 3.0 years with zero as a minimum value for normalized time, evaluate the system for the minimum value ( $K = 1$ ) and maximum values ( $K = \infty$ , where  $K$  is  $> 10^5$ ) of the inverse dormancy factor using steps of 1/10th of a year. Further, when calculating the mean life of the system, the initial value for the upper limit (B) of integration is to be 10.0.

The above problem formulation would be entered into the program by stating that the generalized equation 1 is required. The pertinent data would be inserted by:

```
$VAR; LAMT = 3.0; STEP = 0.1; RV = 1.0; MIN = 0.0;
```

```
5 S = 1; K = 1.0, 100000.0; B = 10.0; OPTION = 2; $END
```

The variable for the family of parameters in this example would be K. Thus the program would serve to evaluate the upper and lower bounds of the system reliability with respect to the inverse dormancy factor (K). A sample run of the program to evaluate the above model formulation is hereinafter provided along with a portion of a typical printout of the computed results and requested off-line graphical plots (Figures 4 and 5).

#### Complex Systems

As earlier mentioned, the equations in the repository of the subject program define basic or primitive systems. Equations representing more complex systems may be readily formulated by combining the basic equations in series reliability with one another.

The description of a complex system is entered by first enumerating the equation numbers of the basic systems. For example, using the namelist VARIABLE notation, "\$VAR 1; PROD = 1, 2; \$END;" states that equation 1 and equation 2 are to be configured in series reliability. The parameter specifications for these equations would then be entered using the namelist VARIABLE notation.

The set of values for any parameter pertaining to a complex system is stored as a matrix. Thus in the general case of parameter (m,n), the "n" refers to the equation involved and the "m" is an internal index for the set of values that would be attempted successively. For example, C(1,2) = 1.0, 0.99 states that in



The semicolons (;) denote carriage returns. The ease and compactness with which complex systems can be specified in the program is demonstrated by the above example.

#### Structural Implementation

5 The foregoing sections described the performance capabilities of the program. This section briefly describes the structural implementation of the present invention.

The program consists of a number of primary subroutines. The interrelationship between these primary subroutines is shown  
10 in the simplified diagram of Figure 1. Generally considered, the overall program has four broadly defined segments which respectively deal with:

- (i) reading in of data and initializing of the logical flow of the program;
- 15 (ii) the functions that are to be performed using the input data;
- (iii) the repository of the general equations that model fault-tolerant systems and the relevant mathematical routines required to elevate these  
20 equations; and
- (iv) initializing output formats, passing the data, and outputting it as 2D plots, 3D projections, or as tables.

As shown by Figure 1, MAIN is the driver for the program and  
25 each of the four segments are under the control of MAIN, which sets the DO loops, determines what and how many times each function is to be performed, and controls the mode in which the results are to be outputted. It is noted that the conventional use of reference numerals has been omitted from Figure 1 in favor of the  
30 computer words or acronyms used to identify the different sub-



routines to avoid unnecessary confusion or complexity that may result from the excessive use of reference indicia.

At the start of a programmed process, MAIN calls READIN to have the subroutine READIN write out questions for the user to answer and record the answers provided. These questions are put in a logical manner with a large number of options to permit the user flexibility in the specification of his problem. A large number of diagnostics and automatic recovery from a user's input errors are provided, i.e., the provision for default values.

Typically, READIN writes out a question, reads in the user's answer to the question, and if the echo feature has been requested, READIN echoes back the answer just read. READIN then calls SCAN passing to it the array containing the information read-in for recognition. SCAN determines whether an answer was a YES or a NO or whether it was a parameter input. If an answer was a parameter input, then SCAN determines its identity. If an input error is detected, the user is asked to try again. READIN thus gathers input data from the user and determines the identity, and order, of subroutines and features that need to be called. The logic of READIN and the decision tree that the user has to traverse is shown in the flowchart illustrated by Figures 2, 3 and 4.

Returning from READIN, MAIN calls SEARCH. SEARCH proceeds to count the number of values that were inputted for each of the system parameters. The number of values counted determines how many times a particular subroutine or function has to be iterated. These values then form the values of the DO limits in the MAIN program. The actual value is obtained by accessing the particular element of the 16 x NPT parameter matrix.

Returning from SEARCH, MAIN asks the user to specify which parameter is to be the family variable. The user's response is

read, optionally echoed back and recognized by SCAN. MAIN then determines which one of three possible parameters -- T, LAMT, or ELAMT -- had been inputted. MAIN then prepares the DO loop limits and rearranges their order in accordance with the inputted family parameter. The initial nested order of the DO loops with respect to the system parameters is LAMBDA, Mu, S, N, K, Q, C, RV, Z, W and P. This initial ordering of the parameters is changed in processing since any of these parameters may be specified to be the family parameter and the innermost DO loop must necessarily correspond to this family parameter. Thus the original position of the parameter selected is interchanged with the innermost parameter, namely P.

MAIN also calls the subroutine RELATE in order to determine the unspecified parameters of the class, LAMBDA, Mu, LAMT, MUT, ELAMT and K. Since these parameters are interrelated, some of them may not have been directly inputted. RELATE readily determines, as necessary, values for those parameters that are unspecified by using the parameter that have been explicitly inputted.

MAIN, using the subroutine RITE, writes the table header for the table of reliability calculations. The header identifies the equation number and the parameters involved. MAIN then calls RELEQS which supplies the desired reliability equation with the necessary parameter values in order to perform the desired reliability calculation. The respective equation subroutines make use of the standard FORTRAN math routines and the math routines provided by the program in accordance with the invention.

Depending on the options read-in by READIN, MAIN then calls upon the subroutines that serve to evaluate the functions to be performed such as the subroutine INTEGER to evaluate the functions MTF and reliability at MTF, etc. Finally, MAIN asks if the user

wishes to specify another parameter as the family parameter. If another family parameter is specified, the data read-in by READIN is retained and, using the new family parameter, MAIN starts its new cycle.

5 Table IV hereinbelow provides a summary of the subroutines that may be used in conjunction with the subject invention. Certain ones of the subroutines are standard library routines as indicated.

TABLE IV

10	<u>Subroutine</u>	<u>Descriptions</u>
	1. MAIN	- reads the inputted parameters, sets up their arrays, sets up the DO LOOPS for their sequencing, and otherwise acts as the driver for the program.
15	2. RELATE	- computes the relationships between $\mu(MV)$ , $K$ , $\lambda(LAMBDA)$ , $\mu T(MVT)$ and $\lambda T(LAMT)$ .
	3. RITE	- writes out headings for tables.
	4. RELEQS	- calls the reliability subroutine specified by the selected equation (NEQ).
20	5. Equation 1A	- description of the general reliability equation of a hybrid-redundant system for $1 \leq K < \infty$ .
	6. Equation 1B	- same as 1A but with $K = \infty$ .
25	7. Equation 2A	- description of the general reliability equation of a standby-replacement system for $1 \leq K < \infty$ .
	8. Equation 2B	- same as 2A but with $K = \infty$ .
	9. Equation 3	- description of equation 3 (void).
30	10. Equation 4A	- description of the reliability equation

TABLE IV Con't.

	<u>Subroutine</u>	<u>Descriptions</u>
		of a Hybrid/Simplex system for $1 \leq K < \infty$ .
5	11. Equation 4B	- same as 4A but with $K = \infty$ .
	12. Equation 5	- description of the equation for a TMR system where the probability of a unit failing to logical one or zero is parameterized.
10	13. Equation 6	- description of the general equation for a simplex system.
	14. Equation 7	- description of equation 7 (void).
	15. SIMPLE	- computes the unreliability, simplex reliability, simple reliability improvement factor (SIMPIF), and simple gain (SIMGAIN).
15	16. READIN	- reads in and checks data for the reliability equations and the plots and writes instructions.
20	17. RIFDIF	- computes the comparative reliability by factors: reliability difference (DIFF), relative improvement factor (RIF), and reliability gain (GAIN).
	18. INTEGER	- computes the system mean life (MTF), and the reliability at the mean life.
25	19. SIMPRI	- computes the comparative reliability factors: maximum mission time (TMAX), simplex maximum mission time (SIMTMAX), and the ratio of these (SIMTIF).
30	20. PARAR1	- computes the comparative reliability

TABLE IV Con't.

Subroutine	Descriptions
5	factors: the ratio of maximum mission times (RATIF) for the various system parameters specified.
21. BISECT	- this subroutine computes $\lambda T(LAMT)$ for given reliability using regula falsi method.
10	22. PLOTT - plots the maximum mission time functions TMAX, SIMTMAX, SIMTIF, and RATIF.
23. EQUAL	- calculates the locus of RV such that the system reliability equals the unit reliability, R.
15	24. PLOTRV - plots the locus of RV such that the system reliability equals the unit reliability, R.
25. AXIS2	- sets up the array containing the values of the family parameter used for 3D plots.
26. PLOTR	- is a driver for the plot routine - KCPLLOT.
20	27. XYGRID - for 2D plots, scales X and Y axis according to the range inputted and also provides automatic scaling.
28. PLOT3D	- is a driver for the 3D plot routines.
25	29. SURF - for 3D plots, contains points for the surface values.
30. SCAN	- scans the array ANSWER for a Y (for YES) or a N (for NO) and for parameter entries L, M, S, N, K, Q, L, R, Z, P or W.
30	31. SEARCH - counts the number of values for each of the inputted variables.

TABLE IV Con't.

	<u>Subroutine</u>	<u>Descriptions</u>
	32. ROWPLT	- labels the plots generated.
5	33. KCPLT	- is a standard plotting routine available in the library of subroutines at the Jet Propulsion Laboratory.
	34. ROMBD	- is a standard numerical integration routine available in the library of subroutines at the Jet Propulsion Laboratory.
10	35. RCOMB	- computes the generalized binomial coefficients (those not necessarily having integer values).
	36. PROD	- calculates special product factors to facilitate the computation of the reliability equations.
15	37. PRODL	- similar to PROD.
	38. INSTR	- is a diagnostic routine to diagnose users YES/NO responses.
	39. FFAC	- computes factorials.
20	40. FNCK	- computes binomial coefficients.
	41. STRT3, ORG3, ADV3, FIN3, PPL, PLOTS, PLOT	- are miscellaneous 3D plot directive routines.

Program Protocol and Sample Run

The following is a sample run of the subject invention in "conversational" mode to illustrate the interaction of the program queries and the user responses and more generally, exemplary results that can be obtained. A portion of a typical printout of the computed results is also provided.

Reference is made to the flow diagrams illustrated by Figures 2, 3 and 4 by the reference numerals in parenthesis,

i.e., (Block 10).

@XQT ATMAN, CARE

HELLO TERMINAL - I AM YOUR RELIABILITY ANALYST WITH THE  
CARE (COMPUTER-AIDED RELIABILITY ESTIMATION) PACKAGE.

5 DO YOU WISH TO HAVE YOUR RESPONSES TO MY  
QUESTIONS PRINTED BACK FOR VERIFICATION (Block 10).  
ANSWER YES OR NO.

YES

10 DO YOU WISH TO KNOW THE DEFINITIONS OF RELIABILITY  
TERMS AND PARAMETERS USED HERE. (Block 12)  
ANSWER YES OR NO.

YES

15 THE DEFINITIONS OF THE VARIOUS RELIABILITY PARAMETERS  
AND TERMS ARE AS FOLLOWS. (Block 14)

T = MISSION TIME.

R = SYSTEM RELIABILITY.

S = THE TOTAL NUMBER OF SPARES.

n =  $(N - 1)/2$  where N is the total # of multiplexed units.

20 K = INVERSE DORMANCY FACTOR =  $(\text{LAMBDA}/\text{MU})$ .

C = COVERAGE FACTOR.

= CONDITIONAL PROBABILITY OF SYSTEM RECOVERING GIVEN A  
FAILURE OCCURANCE

Q = QUOTA, NUMBER OF IDENTICAL UNITS IN A SIMPLEX SYSTEM.

25 W = NUMBER OF CASCADED UNITS.

Z = NUMBER OF IDENTICAL SYSTEMS IN SERIES.

P = PROBABILITY OF A UNIT FAILING TO A LOGIC ZERO.

RV = RELIABILITY OF THE RESTORING ORGAN.

MU = UNPOWERED FAILURE RATE OF A SIMPLEX SYSTEM =  $K/\text{LAMBDA}$ .

30 LAMBDA = POWERED FAILURE RATE OF A SIMPLEX SYSTEM =  $K*\text{MU}$ .

LAMT = NORMALISED TIME = LAMBDA\*MISSION TIME.

ELAMT = EXP(-LAMT).

REL = SYSTEM RELIABILITY.

UNREL = SYSTEM UNRELIABILITY = (1 - REL).

5 SIMREL = SIMPLEX RELIABILITY = ELAMT.

SIMGAIN = GAIN IN RELIABILITY WITH REFERENCE TO A SIMPLEX  
SYSTEM.

= REL/SIMREL.

10 SIMRIF = RELIABILITY IMPROVEMENT FACTOR WITH REFERENCE TO  
A SIMPLEX SYSTEM.

= (1 - SIMREL)/(1 - REL).

DIFF = DIFFERENCE IN RELIABILITIES = R(2) - R(1).

RIF = RELIABILITY IMPROVEMENT FACTOR = (1-R(1))/(1-R(2)).

GAIN = GAIN IN RELIABILITY = R(2)/R(1).

15 SIMTMAX = MAXIMUM MISSION LENGTH OF A SIMPLEX SYSTEM FOR A  
GIVEN MISSION RELIABILITY R1.

TMAX = MAXIMUM MISSION LENGTH OF THE SYSTEM FOR A GIVEN  
MISSION RELIABILITY R1.

20 SIMTIF = TIME IMPROVEMENT FACTOR WITH REFERENCE TO THE  
SIMPLEX SYSTEM

= TMAX/SIMTMAX

RATIF = TIME IMPROVEMENT FACTOR = TMAX(2)/TMAX(1).

DO YOU NEED INSTRUCTIONS FOR RUNNING THE CARE PROGRAM (Block 16)

ANSWER YES OR NO

25 YES

SHORTCOMMENT (block 18) - THE CARE PROGRAM COMPUTES, WITH  
RESPECT TO THE SELECTED EQUATIONS AND PARAMETERS THE FOLLOWING  
RELIABILITY FUNCTIONS - THE RELIABILITY (REL), UNRELIABILITY  
(UNREL), SIMPLEX RELIABILITY (SIMREL), SIMPLE GAIN (SIMGAIN),  
30 SIMPLE RELIABILITY IMPROVEMENT FACTOR (SIMRIF), MEAN TIME



TO FAILURE (MTF), RELIABILITY AT THE MTF, RELIABILITY  
DIFFERENCE (DIFF), RELIABILITY GAIN (GAIN), RELIABILITY  
IMPROVEMENT FACTOR (RIF), SIMPLE MAXIMUM MISSION TIME  
(SIMTMAX), MAXIMUM MISSION TIME (TMAX), SIMPLE TIME  
5 IMPROVEMENT FACTOR (SIMTIF), AND THE RATIO OF TIME  
IMPROVEMENT FACTORS (RATIF).

2D AND SOME 3D PLOTS CAN BE OBTAINED FOR THE ABOVE COMPUTATIONS.  
VARIOUS PLOTTING OPTIONS TO SPECIFY THE ABSCISSA, THE RANGE  
10 OF ABSCISSA AND ORDINATE VALUES ARE AVAILABLE. ABILITY TO  
PLOT 3D INTERSECTIONS OF 3D PROJECTIONS WITH 2D PLANES IS  
ALSO AVAILABLE.

THE CARE PROGRAM ALSO EVALUATES COMPLEX RELIABILITY FUNCTIONS  
15 FORMED BY TAKING PRODUCTS OF THE BASIC RELIABILITY EQUATIONS.

CARE HAS A MAXIMUM OF 7 DIFFERENT RELIABILITY EQUATIONS  
THESE ARE TABULATED BELOW.

1.  $R(N,S) = F(T, \text{LAMBDA}, \text{MU}, S, N, K, \text{RV}, Z, W)$

20 THIS IS THE GENERAL RELIABILITY EQUATION OF AN HYBRID-  
REDUNDANT SYSTEM.

2.  $R(Q,S) = F(T, \text{LAMBDA}, \text{MU}, S, K, Q, C, Z, W)$

THIS IS THE GENERAL RELIABILITY EQUATION OF A STANDBY-  
REPLACEMENT SYSTEM.

25 3. VOID

4.  $H/S(3,S) = F(T, \text{LAMBDA}, \text{MU}, S, K, \text{RV}, Z, W)$

THIS IS THE RELIABILITY EQUATION OF A HYBRID-SIMPLEX SYSTEM

5.  $R(3,O) = F(T, \text{LAMBDA}, \text{RV}, Z, W, P)$

30 THIS IS THE EQUATION FOR A TMR SYSTEM WHERE THE PROBABILITY  
OF A UNIT FAILING TO LOGICAL ONE OR ZERO IS PARAMETERIZED.

6.  $R(1,0) = (\text{EXP}(-\text{LAMBDA} * T)) ** (Z/W)$

THIS IS A GENERAL EQUATION FOR A SIMPLEX SYSTEM.

7. DUMMY

THIS IS A DUMMY EQUATION WHICH IS ALL SET UP TO RECEIVE  
A NEW EQUATION.

INSTRUCTIONS WILL BE GIVEN FOR ENTERING INPUT DATA  
AT THE TIME THE INPUT DATA IS NEEDED BY THE PROGRAM.

DO YOU WISH TO FORM A COMPLEX EQUATION WHICH IS  
THE PRODUCT OF THE PRIMARY EQUATIONS. (Block 20)

ANSWER YES OR NO

NO

TYPE IN COLUMN 1 THE NUMBER OF THE RELIABILITY EQUATION  
TO BE USED - 1 THROUGH 7 (Block 22)

1

INPUT VARIABLES FOR EQUATION 1 (Block 24)

T, LAMT, OR ELAMT MUST BE SPECIFIED AND ITS VALUE

IS THE MAXIMUM VALUE FOR THAT VARIABLE. MIN IS THE MINIMUM  
AND STEP IS THE INCREMENT FOR T, LAMT, OR ELAMT.

SOME VARIABLES THAT ARE NEEDED BY THE EQUATIONS ARE SET  
EQUAL TO A DEFAULT VALUE IF THEY ARE NOT INPUTTED. THESE  
VARIABLES AND THEIR DEFAULT VALUES ARE: S=1, N=1, Z=1, W=1  
Q=1.0D0, C=.999...D0, P=1.0D0, MIN=0.0D0,  
STEP=1.0D0, AND ELAMT=1.0D0.

IF B IS INPUTTED, THEN THIS VALUE IS USED AS THE FIRST  
GUESS FOR THE UPPER LIMIT OF INTEGRATION IN THE CALCULATION  
OF MTF.

IF OPTION-1, THEN DIFF, RIF, AND GAIN ARE CALCULATED FOR

ALL POSSIBLE COMBINATIONS OF THE PARAMETER. IF OPTION=2,  
THEN DIFF, RIF, AND GAIN ARE CALCULATED FOR THE LAST TWO  
PARAMETER VALUES. IF OPTION=0 OR IS NOT INPUTTED, THEN THE  
PROGRAM WILL ASK THE USER AS TO WHICH PARAMETER VALUES  
5 DIFF, RIF, AND GAIN ARE TO BE CALCULATED.  
NOTE: DIFF, RIF, AND GAIN ARE NOT COMPUTED IF THE USER IS  
CALCULATING THE PRODUCT OF RELIABILITIES OR PLOTTING 3-D.  
THE VARIABLES FOR EQUATION 1 ARE INPUTTED USING VAR  
AS THE NAMELIST NAME. A SAMPLE INPUT FOR EQUATION 5 FOLLOWS:

```
10 $VAR  
T=12.0D0,  
LAMBDA=1.0D0,1.5D0,2.0D0,  
RV=1.0D0,  
Z=1,  
15 W=1,6,  
OPTION=2  
B=10.0D0  
$END
```

NOTE: NAMELIST INPUT IGNORES COLUMN 1

20 THE INPUT VARIABLES ARE TYPED AS FOLLOWS

DOUBLE PRECISION: T, LAMT, ELAMT, MUT, LAMBDA, MU,  
K, RV, Q, C, P, MIN, STEP, AND B

INTEGER: S, n, W, Z, AND OPTION

INPUT VARIABLES NOW (Block 26)

25 INPUT VARIABLES FOR EQUATION 1

BEGIN TYPING IN COL 2 USING \$VAR...\$END NAMELIST DELIMITERS.

DO YOU WISH TO MAKE ALTERATIONS TO THE \$VAR LIST

ANSWER YES OR NO (Block 28)

NO

30 DO YOU WISH TO HAVE 2-D RELIABILITY PLOTS - ANSWER YES

OR NO(Block 30)

YES

INPUT A 1 IN THE COLUMN SPECIFIED BELOW IF YOU WISH (Block 32)  
THE CORRESPONDING PLOT OPTION. OTHERWISE INPUT 0.

5 NOTE: WHEN PERFORMING PRODUCT OF RELIABILITIES, NO OTHER  
PLOT OPTION BESIDES PRODUCT OF RELIABILITIES MAY BE SPECIFIED.

COLUMN 1 - PLOTS PRODUCT OF RELIABILITIES

COLUMN 2 - PLOTS RELIABILITY

COLUMN 3 - PLOTS DIFF, RIF, AND GAIN

10 COLUMN 4 - PLOTS MTF AND RELIABILITY AT MTF

COLUMN 5 - PLOTS UNRELIABILITY

01100

FOR ABSCISSA, INPUT 1 IN COLUMN 1 IF ABSCISSA IS T, (Block 34)

1 IN COLUMN 2 IF ABSCISSA IS  $\log(T)$  - BASE 10,

15 1 IN COLUMN 3 IF ABSCISSA IS  $\lambda MT$ ,

1 IN COLUMN 4 IF ABSCISSA IS  $\log(\lambda MT)$  - BASE 10,

1 IN COLUMN 5 IF ABSCISSA IS  $\exp(-\lambda T)$ ,

1 IN COLUMN 6 IF ABSCISSA IS  $\log(\exp(-\lambda MT))$  - BASE 10.

\*\*1\*\*\*

20 IF YOU WISH TO PLOT A CERTAIN RANGE OF X-AXIS VALUES (Block 36)

FOR THE 2-D PLOTS, ENTER LEFT-END POINT IN COLUMNS 1-8 WITH  
FORMAT F8.0 AND RIGHT-END POINT IN COLUMNS 9-16 WITH FORMAT  
F8.0;

OTHERWISE INPUT NO

25 NO

IF YOU WISH TO PLOT A CERTAIN RANGE OF Y-AXIS VALUES (Block 38)

FOR THE 2-D PLOTS, ENTER LEFT-END POINT IN COLUMNS 1-8 WITH  
FORMAT F8.0 AND RIGHT-END POINT IN COLUMNS 9-16 WITH FORMAT  
F8.0;

30 OTHERWISE INPUT NO

NO

DO YOU WISH TO PLOT THE LOCUS OF RV SUCH THAT THE (Block 40)  
SYSTEM RELIABILITY EQUALS THE UNIT RELIABILITY.

ANSWER YES OR NO

5

NO

DO YOU WISH TO HAVE 3-D RELIABILITY PLOTS - ANSWER YES OR  
NO (Block 42)

NO

DO YOU WISH TO CALCULATE MAXIMUM MISSION TIME AND SIMPLE  
TIME (Block 44)

10

FOR GIVEN RELIABILITY - ANSWER YES OR NO

YES

DO YOU WANT PLOTS FOR THESE CALCULATIONS - ANSWER YES OR  
NO (Block 46)

15

YES

DO YOU WISH TO CALCULATE MAXIMUM MISSION TIME FOR (Block 48)  
GIVEN RELIABILITY AND COMPARE IT AGAINST OTHER PARAMETERS  
ANSWER YES OR NO

YES

20

INPUT IN COLUMN 1 ONE OF THE FOLLOWING THREE OPTIONS: (Block 50)

1. MAXIMUM MISSION TIME IS COMPARED AGAINST ALL POSSIBLE  
COMBINATIONS OF THE PARAMETER,

2. MAXIMUM MISSION TIME IS COMPARED AGAINST THE LAST TWO  
PARAMETER VALUES,

25

3. THE PROGRAM ASKS THE USER AS TO WHICH PARAMETER VALUES  
MAXIMUM MISSION TIME IS TO BE COMPARED.

1

DO YOU WANT PLOTS FOR THESE CALCULATIONS - ANSWER YES OR  
NO (Block 52)

30

NOTE: WHEN EXERCISING OPTION 1, THE PROGRAM PLOTS

ONLY THE FIRST 15 PARAMETER COMPARISONS

YES

INPUT THE FOLLOWING 4 VARIABLES EACH WITH FORMAT F8.0 (Block 54)

COLUMNS 1-8 - REFERENCE RELIABILITY R2

5 COLUMNS 9-16 - MINIMUM RELIABILITY R1

COLUMNS 17-24 - MAXIMUM RELIABILITY R1

COLUMNS 25-32 - RELIABILITY R1 STEP SIZE

1.000 .000 1.000 .100

DO YOU WISH TO HAVE PRINTED TABLE OF RELIABILITY RESULTS

10 (Block 56)

ANSWER YES OR NO

YES

DO YOU WISH TO HAVE PRINTED TABLE OF DIFF, RIF (Block 58)

AND GAIN RESULTS - ANSWER YES OR NO

15 YES

DO YOU WISH MTF AND RELIABILITY AT MTF RESULTS PRINTED

(Block 60)

ANSWER YES OR NO

YES

20 DO YOU WISH PRINTED RESULTS OF THE MAXIMUM MISSION (Block 62)

TIME CALCULATIONS - ANSWER YES OR NO

YES

TYPE IN THE VARIABLE THAT IS TO BE USED

FOR THE FAMILY OF PARAMETERS - MUST BE SPECIFIED

25 K

Following is an exemplary portion of a printout that is  
generated by the program in accordance with the invention

30

CALCULATIONS FOR EQUATION 1A (NI MEANS NOT INPUTTED)

PARAMETER IS K

LAMBDA	MU	S	n	K	Q
NI	.0000000	1	1	.1000000+01	NI

5

C	RV	Z	W	P	MUT
NI	.1000000+01	1	1	.1000000+01	NI

LAMT	REL	UNREL	SIMREL	SIMGAIN	SIMRIF
.000	1.0000000	.0000000	1.0000000	.1000000+01	.1000000+36
10 .100	.9967989	.0032011	.9848374	.1101633+01	.2972798+02

3.000	.0139037	.9860963	.0497871	.2792626+00	.9636107+00
-------	----------	----------	----------	-------------	-------------

15 MEAN TIME TO FAILURE - MTF = .10833333+01

UPPER LIMIT FOR INTEGRATION - 0 = .15000000+02

RELIABILITY AT MTF = .41653059+00

MAXIMUM MISSION TIME REFERENCE R2 = 1.00000

R1	SIMLAMTMAX	LAMTMAX	SIMTIF
.00000	INFINITY	INFINITY	.1000000+01
.10000	.2302585+01	.1948467+01	.8462084+00
.20000	.1609438+01	.1549781+01	.9629332+00

25

1.00000	.0000000	.0000000	.1000000+01
---------	----------	----------	-------------

TMAX AND SIMTIFF PLOT COMPLETED

30

CALCULATIONS FOR EQUATION 1B (NI MEANS NOT INPUTTED)

PARAMETER IS K

LAMBDA	MU	S	n	K	Q
NI	NI	1	1	NI	NI
C	RV	Z	W	P	MUT
NI	.1000000+01	1	1	.1000000+01	NI

5

LAMT	REL	UNREL	SIMREL	SIMGAIN	SIMRIF
.000	1.0000000	.0000000	1.0000000	.1000000+01	.1000000+36
.100	.9975401	.0024599	.9048374	.1102452+01	.3868510+02
.200	.9838134	.0161866	.8187307	.1201632+01	.1119870+02
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.

10

15

MEAN TIME TO FAILURE - MTF = .11666667+01

UPPER LIMIT FOR INTEGRATION - B = .15000000+02

RELIABILITY AT MTF = .41978696+00

20

MAXIMUM MISSION TIME REFERENCE R2 = 1.00000

R1	SIMLAMTMAX	LAMTMAX	SIMTIF
.00000	INFINITY	INFINITY	.1000000+01
.10000	.2302585+01	.2083571+01	.9048836+00
.20000	.1609438+01	.1666156+01	.1035241+01
.	.	.	.
.	.	.	.
.	.	.	.
.90000	.1053605+00	.4224357+00	.4009430+01
1.00000	.0000000	.0000000	.1000000+01

25

30



TMAX AND SIMTIF PLOT COMPLETED

MAXIMUM MISSION TIME FOR K = .1000000+001

AND K = .1000000+006 FOLLOWS FOR EQUATION 1B

5 REFERENCE R2 = 1.00000

R1	TMAX1	TMAX2	RATIF
.00000	INFINITY	INFINITY	.1000000+01
.10000	.1948467+01	.2083571+01	.1069339+01
.20000	.1549781+01	.1666156+01	.1075091+01

.90000	.3862209+00	.4224357+00	.1093767+01
1.00000	.0000000	.0000000	.1000000+01

15 1 MAXIMUM MISSION TIME PLOTS FOR VARYING  
PARAMETER VALUES COMPLETED

DIFF, RIF, AND GAIN FOR K = .1000000+001

AND K = .1000000+006 FOLLOWS FOR EQUATION 1B

LAMT	DIFF	RIF	GAIN
.00000	.00000	INFINITY	.100000+01
.10000	.741191-03	.130131+01	.100074+01
.20000	.439928-02	.127178+01	.100449+01
.30000	.110269-01	.124462+01	.101168+01

3.00000	.519645-02	.100530+01	.137375+01
---------	------------	------------	------------

DO YOU WISH TO SPECIFY ANOTHER PARAMETER

ANSWER YES OR NO

NO

QFIN

5 Sample plots of the above computed data for Reliability (REL) and Difference in Reliability (DIFF) as a function of maximum normalized mission time (LAMT) are provided by Figures 5 and 6, respectively.

10 From the foregoing, it is now apparent that the subject program makes available a highly flexible means for obtaining computer-aided estimates of reliability with respect to specific model formulations. More specifically, it is now clear that the subject program offers the advantages of being able to be operated in a "conversational" or batch mode, providing a multiplicity of reliability functions applicable to all equations  
15 maintained in an independent repository, permitting any complex model to be formulated by combining basic equations in the repository, and providing a repository that is extendable.

20 While a preferred embodiment of the present invention has been described hereinabove, it is intended that all matter contained in the above description and shown in the accompanying drawings be interpreted as illustrative and not in a limiting sense and that all modifications, constructions and arrangements which fall within the scope and spirit of the present invention may be made.