A RELIABILITY/AVAILABILITY SIMULATION MODEL FOR EVALUATING NETWORK SYSTEMS

Raymond John Jenkins

A project report submitted to the Faculty of Engineering, University of the Witwatersrand, Johannesburg, in partial fulfilment of the requirements for the degree of Master of Science in Engineering.

Johannesburg, 1992

DECLARATION

I declare that this project report is my own, unaided work. It is being submitted for the Master of Science in Engineering at the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination at any other University.

RJJenkins

(R.J. Jenkins)

26th day of October 1992

ABSTRACT

The simulator uses the Monte Carlo technique to quickly and accurately estimate the reliability and availability of complex network systems. Non-exponential failure and repair distributions are included in the model, as is standby redundancy and K out of N active redundancy. The program is easy to use and will work on a large variety of computers and FORTRAN compilers. Some knowledge of FORTRAN is required to program the simulator for each reliability network. The simulator is limited to the analysis of network systems, i.e. those systems whose logic can be fully represented by a reliability block diagram. The applicability of the model was demonstrated by the analysis of numerous systems in the aerospace and industrial environments. Validation of the model was accomplished by comparing these results with analytically determined values, or those from **AMIR**® and **SPAR**® where an analytic solution was impossible.

CONTENTS

LIST OF FIGURES

Figure:

LIST OF TABLES

NOMENCLATURE

MTBF	-	Mean Time Between Failure
MTTR	-	Mean Time to Repair
A_i	-	Inherent Availability
A_o	-	Operational Availability
A_a	-	Achieved Availability
EWS	-	Electronic Warfare System
CM	-	Corrective Maintenance
PM	-	Preventive Maintenance
LDT	-	Logistic Delay Time
RBD	-	Reliability Block Diagram
FOM	-	Figure Of Merit

1 INTRODUCTION

Reliability and availability have become important criteria in the design and operational phase of systems. For example, the operational defects of an electronic warfare system are critical to a combat helicopter completing its mission. The safety aspects of a Space Shuttle computer system are vital to the survival of the crew in space. The unavailability of an industrial system causes a loss in production.

Design engineers are now more than ever required to perform trade-off studies between system availability, reliability, technical performance and life cycle cost. The engineer therefore requires an accurate, economical and easy to use system reliability and availability estimation tool. Unfortunately, the analytical estimation of system reliability and availability becomes difficult and expensive even for the most simplest of systems. In fact, the analytic approach is often inadequate for most engineering needs.

The alternative approach is to simulate system failures and repairs using the Monte Carlo technique. This technique entails the generation of component random times to fail and repair from which the system time to fail and repair can be determined. The system failure and repair time obtained in this manner must be viewed as the outcome of an experiment. This experiment is then repeated many times until an adequate estimate of system reliability and availability is obtained.

Forry[13] has carried out extensive work in the development of Monte Carlo simulation models for estimating large scale system reliability and availability. His work was primarily directed at the modelling of network systems.

Goldfeld and Dubi[14] have addressed the reliability and availability analysis of general systems using the Monte Carlo technique. General systems are non-network type systems, i.e. their logic cannot be fully represented by a Reliability Block Diagram. Their work led to the development of commercially available Monte Carlo based system engineering software. This software is currently available in two packages, i.e. **AMIR®** and **SPAR®**. Both packages are suitable for reliability and availability analyses. **SPAR®** can model multiple systems at the same time as well as taking into account the effects of spare part shortages.

The reliability/availability simulation model developed in this report (also referred to as the simulator) is based to a large extent on the work carried out by Forry[13] with the inclusion of the system function approach used by Dubi[9] to find system failure times. Forry[13] used a complex PERT algorithm to calculate system failure times from the component failure times. Dubi[9] on the other hand found the system failure time by checking the status of the system at each stochastic event. The system status is a function of the status of each component which is defined by the system function.

Using the simulator, reliability and availability estimates are easily obtained for systems arranged in different configurations. Standby redundancy and K out of N active redundancy are easily included in the model, as are non-exponential failure distributions and repair distributions. The model also allows one to change the number of repair teams and select between either leaving components on or switching them off during system repair. The simulation model is however limited to network systems only, i.e. those systems whose logic can be fully represented by a Reliability Block Diagram.

Real life systems are complex and we will never be able to model the system exactly as it is in real life. Approximations can however be made which will not affect the model results significantly. It is up to the engineer to make these approximations and establish whether Monte Carlo simulation is in ~~fact~~ required to solve the problem.

For some systems, the chances of system failures occurring during a certain time interval are extremely remote. For example, the unreliability of a quadruplex flight control computer system may be one catastrophic failure in 100 million flights. Millions of simulation histories are therefore required before such an event is actually seen. Unfortunately, it is often impractical to run millions of histories due to computer time limitations. This is a serious disadvantage of the Monte Carlo method. Goldfeld and Dubi[14] overcame this problem by enhancing the probability of rare events and then compensated the final result to ensure an unbiased solution. This technique, often referred to as a biasing technique, was not included in this study.

The applicability of the simulator was demonstrated by the analysis of five systems in the aerospace and industrial environments. The results of the model were validated by analytic means where possible and by **SPAR®** or **AMIR®** where an analytic solution was not practical. Some common definitions of reliability and availability have been discussed in this report as they have always been a source of confusion. It is important for the user to understand the logic of the simulator and to be able to distinguish between a network system and a general system. A detailed discussion has therefore been included on these two topics. A brief description can also be found of the simulation program which consisted of a main program and several subroutines.

This report presumes that the reader is familiar with basic reliability theory and detailed explanations of underlying theory have therefore been avoided. Some theory, applicable to the simulation code, has been included where it was felt necessary.

1.1 COST-EFFECTIVENESS FIGURES OF MERIT

Availability and reliability are only two of the many ingredients which make up a cost-effective system. It is therefore important to put these two parameters into perspective with respect to overall system cost-effectiveness.

Blanchard and Fabrycky[3] state that the basic design objective is to develop a system that will perform its intended function in a cost-effective manner, i.e. do the job effectively at the lowest overall life cycle cost. Some organisations also consider revenues and profits along with cost in their design objective.

Accomplishing this cost-effective design objective requires an optimum balance between criteria such as technical performance, availability, dependability and life cycle cost.

- Technical performance or capability relates to how well the system will perform in the mission environment, i.e. the design adequacy of the system.

- Availability or operational readiness relates to whether the system will be ready to perform its mission when called to do so.

- Dependability or mission reliability relates to whether the system will continue to perform for the duration of the mission, given that it was available to start the mission. Reliability is therefore a measure of the dependability of a system.

The prime ingredients of cost-effectiveness are illustrated in Figure 1.1.1.

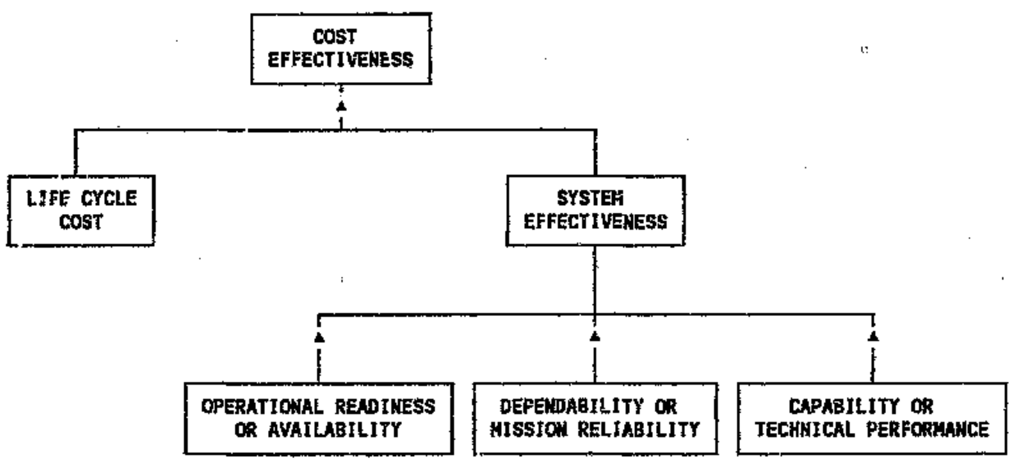

Figure 1.1.1 The Elements of Cost-Effectiveness

Figures Of Merit (FOM's) usually represent a combination of the above system parameters. One would typically employ FOM's such as:

$$FOM = \frac{SYSTEM\ EFFECTIVENESS}{LIFE\ CYCLE\ COST} \tag{1.1.1}$$

$$FOM = \frac{AVAILABILITY}{LIFE\ CYCLE\ COST} \tag{1.1.2}$$

These FOM's are often presented as delta values which allows one to compare alternative systems on the basis of the relative merits of each. Given two or more alternative designs evaluated in a consistent manner, one can select the best based on these delta values.

1.2 RELIABILITY

Watras[23] defines reliability as the probability that an item will perform as required, under stated conditions, for a stated period of time. When an item no longer performs as required we say it has failed. Caplen[5] states that a failure is the termination of the ability of an item to perform its required function.

The engineer must construct an appropriate reliability model from the physical system to model the system requirement. The reliability model will change as the system requirements change. For example, estimating the probability of an aircraft successfully completing its mission and the probability of an aircraft not crashing during the mission require different reliability models for the same physical system. Some of the more common reliability definitions encountered in the aerospace environment are described below.

1.2.1 SAFETY RELIABILITY

Safety reliability is the probability of being able to perform a given mission without any failures or defects that will have a catastrophic effect.

The system requirement would therefore be for the aircraft to survive a mission and system failure would result in the loss of an aircraft and the possible death of the occupants.

The chances of this occurring during a typical flight are usually of the order of one in 10 million for military aircraft and one in 100 million for commercial aircraft. Note, these figures include all systems on the aircraft.

1.2.2 MISSION RELIABILITY

Fielding and Meng[12] define mission reliability as the probability that an aircraft will be able to perform a given mission without any failures or defects that will have an operational effect.

The system requirement would therefore be to accomplish the mission and a system failure would result in the mission being aborted and the aircraft returning to base. Mission reliability performance is often difficult to predict as it depends on what are considered to be defects which impair a mission.

Note, mission reliability and safety reliability are both point estimates of reliability for a specific mission time however the system requirements are not the same.

1.2.3 OVERALL DEFECT RATE

The overall defect rate is the rate at which defects occur in the system. The system as a whole does not necessarily fail at the same rate. Fielding and Meng[12] state that the overall defect rate is the sum of all the component failure rates of the system, i.e.

$$\lambda_{overall} = \lambda_1 + \lambda_2 + \lambda_3 + \ \ldots\ldots \qquad (1.2.3.1)$$

Engineers often invert the overall defect rate and call this the "MTBF" of the system. Statistically speaking, the "MTBF" describes the mean of an exponential failure distribution. Therefore, for this to be mathematically correct all components must have exponential failure distributions and each component failure must cause a system failure. The assumption that the system exhibits an exponential failure distribution has some surprising implications, i.e. the most probable time interval between failures is zero and not the mean as one would expect, also 63 % of all failures would have occurred before the mean life is reached. This "MTBF" is often given many names, i.e. basic reliability, compounded reliability, maintenance reliability, etc.

Evans[10] explains that the acronym "MTBF" is often the cause of difficulties in contracting for reliability. The difficulties range from not understanding the implications of the mathematical assumptions to proving one did or did not obtain the contracted value. For these reasons it is more meaningful for the non-statistician to speak of the rate at which defects occur in the system, which almost any manager or engineer can readily understand, eg, 1 % failures per month. Most non-statisticians are just using "MTBF" as the reciprocal of the overall defect rate anyway, so why not just use the defect rate in the first place. During reliability growth, managers and engineers are concerned with estimating the current reliability that has been achieved, not with calculating some average reliability over the past. Evans[10] explains further that there is a big difference between a failure, a removal and a corrective repair action. The data on defects probably do not, and can not, distinguish adequately between these three concepts. A good rule of thumb which can be applied in this situation is that the removal rate is about twice the failure rate.

Fielding and Meng[12] explain that the overall defect rate is also a good measure of the maintenance effort required to keep the aircraft flying. The reason for this is that each component failure no matter how minor will have to be repaired at some stage. It is interesting to note that for an active redundant system, the redundancy would have improved the mission and/or safety reliability, but the addition of the extra components would have increased the overall defect rate.

1.2.4 DISPATCH RELIABILITY

Fielding and Hussain[11] state that dispatch reliability is the probability of an aircraft departing on time on revenue-earning flights. For large commercial aircraft this is given by:

$$\text{Dispatch Reliability (\%)} = 100 - \frac{\text{No. of delays} > 15\text{ min} + \text{cancellations}}{100\text{ departures}} \qquad (1.2.4.1)$$

1.3 AVAILABILITY

Availability is defined as the probability that a system will be in an operable and committable state, at the start of any prescribed mission, when the mission is called for at a random point in time. Availability does not refer to being able to perform satisfactorily throughout the mission. This issue is addressed by the measures of dependability and reliability.

Bernstein[2] explains that availability uses the operational demand time as the basis for computation, i.e. the time that there is a demand for the system to actually work. This time would therefore exclude time such as weekends, off-duty periods, free time, etc.

When non-operational times are included, e.g. standby, the basis for computation becomes total calendar time and the concept of availability is replaced by operational readiness. Consider a fighter aircraft, it performs sporadic missions and spends most of its time on standby. Now according to convention, the standby time would be excluded from the computation of availability but included in the computation of operational readiness.

Availability can be measured as an average availability or a point availability. Caplen[6] explains that the average availability is measured over the whole duty period whereas the point availability is calculated at a specific point in time. For example, an average availability of 0.8 means that the system is in a condition to work satisfactorily for 80 % of the time. The probability that the system will be available for use at say 10 a.m. today is a point availability. The simulator calculates average availability.

Watras[23] states that the most basic description of average availability is the ratio of system uptime over the total time for which there is a demand for the system, i.e.

$$A = \frac{UPTIME}{UPTIME + DOWNTIME} \qquad (1.3.1)$$

Depending on the type of system being analyzed and on how we wish to measure availability, system states can be assigned to either uptime or downtime. For example, one could say that uptime for a fighter aircraft is sortie time and standby time whereas uptime for a production line is operating time only.

Table 1.3.1 shows four availability measurements which are often found in the aerospace and production environments. The applicable system states for each availability measurement have been shaded for illustrative purposes. It should be noted that for inherent, achieved and operational availability, standby time has been excluded from the computation. Standby is however included in the utilisation factor as downtime. The equipment can either be in a condition to work or be working (internally operational), or it can be failed (internally non-operational). Each availability measurement shown in **Table 1.3.1** is discussed further in the sections which follow.

Table 1.3.1 Availability Measurements

AVAILABILITY MEASUREMENTS	UPTIME AND DOWNTIME ALLOCATIONS				
INHERENT AVAILABILITY	UP		DOWN		
ACHIEVED AVAILABILITY	UP		DOWN	DOWN	
OPERATIONAL AVAILABILITY	UP		DOWN	DOWN	DOWN
UTILISATION FACTOR	UP	DOWN	DOWN	DOWN	DOWN
SYSTEM STATES	OPERATION	STANDBY	CM	PM	LDT
	SYSTEM INTERNALLY OPERATIONAL		SYSTEM INTERNALLY NON-OPERATIONAL		

Notes:

CM - Corrective Maintenance
PM - Preventive Maintenance
LDT - Logistic Delay Time

1.3.1 INHERENT (INTRINSIC) AVAILABILITY

Inherent availability A_i is a conventional indicator of hardware supportability, the measure rises as reliability or maintainability increase and the converse also applies. The measurement covers corrective maintenance but excludes preventive maintenance and delay times such as waiting for spares and repair personnel. One therefore assumes a repairable system operating in an ideal environment where support equipment, tools, skilled manpower, manuals, spares, and repair parts are in abundance.

Watras[23] states that inherent availability is a function of system design only and neglects the effects of supply support in describing system availability. Inherent Availability is useful when evaluating one proposed system against another on the basis of system design performance. Inherent availability can be thought of as an upper bound when determining operational availability. The value of operational availability will approach the value of inherent availability as the supply support posture improves and the supply response time approaches zero.

1.3.2 ACHIEVED AVAILABILITY

This measure is more appropriate for systems with significant mechanical content, i.e. the system undergoes preventive maintenance. Achieved availability A_a covers corrective and preventive maintenance but assumes a perfect support system. By comparing A_i and A_a it is possible to see how effective preventive maintenance is.

1.3.3 OPERATIONAL AVAILABILITY

Operational availability A_o is the practical parameter of availability. It includes preventative and corrective maintenance and all delay times, i.e. waiting for spares and manpower etc. It is the value which can be expected under actual operating conditions for continuous utilisation.

Watras[23] stresses that operational availability goals and thresholds must be considered throughout the system life cycle. These goals are to be defined in the system conceptual and definition phases and used as guidelines throughout the system design and development phase. Once a system becomes operational, A_o based on actual field data, should be used as a basis for ongoing logistic management review and improvement actions.

Sparrius[21] states that if a system's inherent availability is poor, then it should be redesigned. If a systems operational availability is poor and its inherent availability is good then the support system should be redesigned.

1.3.4 UTILIZATION FACTOR

Caplen[6] noted that availability can also be expressed as a utilization factor by defining the time that the system is in standby as downtime.

This measurement is typically found in a continuous production environment where one is trying to achieve the maximum utilisation from equipment. The measurement is more general than the previous three as it includes the time that the equipment could have been used by the operator.

1.4 NETWORK SYSTEMS

Billinton and Allan[4] explain that the reliability of a system can be frequently represented by a network in which the system components are tied together either in a series, parallel or meshed configuration, such as the system shown in **Figure 1.4.1.**

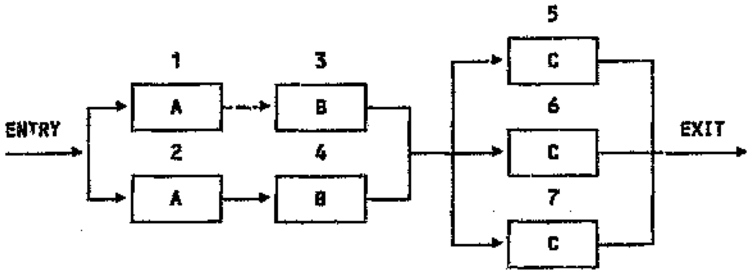

Figure 1.4.1 A Network Type System

Dubi[9] says that if this system is a network system, then all the information concerning the structure of the system is contained in the above figure. The single logical rule being that the system is up as long as their is at least one tie from the entry point to the exit point of the system. A tie is a series of connected active operational components. Dubi[9] explains further that any system which does not follow the above rule is a general system.

A typical example of general system is a fly-by-wire flight control system. The required safety target of one failure in 100 million flights requires the use of a quadruple redundant flight control computer system which includes a voting process. The failure of the system could be either due to failures of components or due to a malfunction in the decision of the voting system. The former type of failure can be easily modelled with the use of a reliability block diagram, however the latter failure contains complex logic which cannot be represented by a reliability block diagram. The system can therefore be categorised as a general system.

It is vital that the relationship between the physical system and its network model be understood before considering any techniques to evaluate these networks. It must be appreciated that the actual system and the reliability network used to model the system may not necessarily have the same topological structure. The reliability network may also change when the requirements of the physical system change. For example, the reliability network of a system is different if the requirement is the survival of the aircraft or the completion of a mission. The physical topology of the system remains the same in both cases.

The simulator only models network type systems and would require extensive modification to model a general system. Note, a reliability network is often referred to as a Reliability Block Diagram (RBD).

2 GENERAL DESCRIPTION

2.1 SUMMARY OF THE MONTE CARLO TECHNIQUE

A simulation model seeks to "duplicate" the behaviour of the system under investigation by studying the interactions between its components. The output of the simulation model is normally presented in terms of selected measures that reflect the performance of the system. For example, one may wish to measure the average time the system spends in the failed state or the rate at which system failures are occurring.

A simulation experiment differs from a regular laboratory experiment in that it can be totally conducted by the computer. By expressing the interactions among the components of the system as mathematical relationships, we are able to gather the necessary information in much the same way as observing the real system (subject of course to the simplifications assumptions built into the model). The simulation allows greater flexibility in representing complex systems that are normally difficult to analyze by standard mathematical models. The Monte Carlo method is based on the general idea of using sampling to estimate a desired result. The sampling process requires the description of the problem by appropriate probability distributions from which samples are drawn.

Forry[13] explains that in the Monte Carlo technique as applied to the simulator, one assumes that the time to fail and time to repair probability distributions are known for each component of the system. It is further assumed that the relationship between component failure and system failure is known and can be described in the form of a Reliability Block Diagram.

Uniformly distributed random numbers are generated and used to determine component times to fail and component times to repair. These component times to fail or repair are then used to determine the system time to fail and repair. The set of system times to fail or repair must be viewed as a random sample of the distribution of system failure times or repair times. Therefore, the data must be operated on in the same manner that real test data would be to determine the form and parameters of the system reliability and availability functions.

2 GENERAL DESCRIPTION

2.1 SUMMARY OF THE MONTE CARLO TECHNIQUE

A simulation model seeks to "duplicate" the behaviour of the system under investigation by studying the interactions between its components. The output of the simulation model is normally presented in terms of selected measures that reflect the performance of the system. For example, one may wish to measure the average time the system spends in the failed state or the rate at which system failures are occurring.

A simulation experiment differs from a regular laboratory experiment in that it can be totally conducted by the computer. By expressing the interactions among the components of the system as mathematical relationships, we are able to gather the necessary information in much the same way as observing the real system (subject of course to the simplifications assumptions built into the model). The simulation allows greater flexibility in representing complex systems that are normally difficult to analyze by standard mathematical models. The Monte Carlo method is based on the general idea of using sampling to estimate a desired result. The sampling process requires the description of the problem by appropriate probability distributions from which samples are drawn.

Forry[13] explains that in the Monte Carlo technique as applied to the simulator, one assumes that the time to fail and time to repair probability distributions are known for each component of the system. It is further assumed that the relationship between component failure and system failure is known and can be described in the form of a Reliability Block Diagram.

Uniformly distributed random numbers are generated and used to determine component times to fail and component times to repair. These component times to fail or repair are then used to determine the system time to fail and repair. The set of system times to fail or repair must be viewed as a random sample of the distribution of system failure times or repair times. Therefore, the data must be operated on in the same manner that real test data would be to determine the form and parameters of the system reliability and availability functions.

2.2 COMPONENT FAILURE AND REPAIR DENSITIES

The most commonly encountered component failure and repair distributions are the negative exponential, normal, lognormal and Weibull distributions. Lengthy statistical descriptions of each of these distributions have not been included in this report as they can be easily found in many statistical texts. Instead, the relevant equations have been presented together with a brief practical discussion.

The distribution of times to failure is of the negative exponential form if the failure rate is constant. In other words, the probability of failure remains the same irrespective of the age of the component. The failure probability density function f(t) and the reliability function R(t) are defined as:

$$f(t) = \frac{1}{MTBF} \exp(-\frac{t}{MTBF}) \qquad (2.2.1)$$

and

$$R(t) = \exp(-\frac{t}{MTBF}) \qquad (2.2.2)$$

Where t is a possible repair time and the MTBF is the life at which 63 % of the components would have failed. The practical significance of this is that components must have working lives much shorter than their mean life.

The exponential distribution is suitable for describing the lifetimes of components whose failure times are not age related, i.e. most electronic components. Nowlan and Heap[18] explain that for complex items, i.e. those with many different failure modes, the failure ages for the component as a whole are usually widely dispersed and are unrelated to a specific operating age. This is a unique characteristic of a complex item. Therefore, most complex mechanical components will exhibit an exponential failure distribution.

Very often in practise, the MTBF is simply estimated by dividing the total hours of all the items by the total number of items failed during that time. Evans[10] gives a good rule of thumb for estimating component MTBF, i.e. the component removal rate is approximately twice the failure rate.

Nowlan and Heap[18] state that for a simple item, i.e. those items with a single or dominant failure mode, the failure ages tend to concentrate about an average age. These components therefore exhibit an age related type of failure. In such cases the distribution of times to failure is often found to follow the normal distribution. The density function f(t) and the reliability function R(t) are defined as:

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp[-\frac{1}{2}(\frac{t-\mu}{\sigma})^2] \qquad (2.2.3)$$

and

$$R(t) = \int_0^t \frac{1}{t\sigma\sqrt{2\pi}} \exp[-\frac{1}{2}(\frac{t-\mu}{\sigma})^2]\ dt \qquad (2.2.4)$$

Where t is a possible time to failure, μ is the mean of the values of t and σ is the standard deviation of t about the mean.

The mean life μ is the life at which 50 % of all components would have failed. The area under the f(t) curve from $-\sigma$ to $+\sigma$ includes 68 % of all failures. The area from -2σ to $+2\sigma$ includes 95 % of all failures and the area from -3σ to $+3\sigma$ includes 99 % of all failures. Therefore, practical speaking all failures are included within 3 standard deviations. The smaller the standard deviation the more the values are clustered around the mean.

Smith and Babb[20] state that for maintenance activities, active repair times are usually distributed according to the log normal rule, i.e. the logarithms of the times to repair are normally distributed. Maintainability is defined as the probability that a failed item will be repaired in time t. The maintainability function M(t) is defined as:

$$M(t) = \int_0^t \frac{1}{t\sigma\sqrt{2\pi}} \exp[-\frac{1}{2}(\ln\frac{t-\mu}{\sigma})^2]\ dt \qquad (2.2.5)$$

Where t is a possible repair time, μ is the mean of the values of ln(t) and σ is the standard deviation of ln(t) about the mean.

In practise it is often found that an abberation occurs in the lognormal distribution of maintenance times, i.e. a secondary peak exists. The reasons for this could include false timekeeping, overmanning and unskilled crews on some jobs. If the secondary peak is very large and approaches the size of the mode, it probably indicates that there are two distinct types of maintenance work represented in the curve. Each of which may have a lognormal distribution of its own.

The two parameter Weibull distribution has the great advantage of being able to fit many life distributions by adjusting the distribution parameters. The density function f(t) and the reliability function R(t) are defined as:

$$f(t) = \frac{\beta}{\eta^{\beta}} t^{\beta-1} \exp[-(\frac{t}{\eta})^{\beta}] \qquad (2.2.6)$$

and

$$R(t) = \exp[-(\frac{t}{\eta})^{\beta}] \qquad (2.2.7)$$

Where t is a possible repair time, η is the characteristic life of the values t and β is the shape parameter of the distribution. The characteristic life is the life at which 63 % of the population would have failed.

O'Connor[19] shows that when the shape parameter β is one the exponential distribution results (constant failure rate), when it is less than one a decreasing failure rate distribution results and when it is larger than one an increasing failure rate distribution results. At a value of 3.5 the distribution approximates the normal distribution. Higher values also produce a distribution which does not depart markedly from the normal distribution. The nomenclature describing the scaling parameter varies from text to text. The characteristic life η was chosen as the scaling constant for the simulator as this is the constant used on commercially available Weibull graph paper.

Smith and Babb[20] state that passive repair times are often described by the Weibull distribution, where the shape and scale factors can be easily found by a graphical analysis of repair times.

It is interesting to note that the MTBF of the exponential distribution, the characteristic life η of the Weibull distribution and $\mu + 0.33\sigma$ of the normal distribution all represent the life at which 63 % of components would have failed.

2.3 SAMPLING COMPONENT FAILURE AND REPAIR DENSITIES

A combination congruential generator was used to generate a sequence of uniformly distributed random numbers within the interval [0,1] as described by Lewis and Orav[17]. The generator is particularly applicable for small computer word sizes and for very long cycle lengths. To produce a uniform [0,1] variate, U_{i+1}, we require the output from three separate congruential generators:

$$X_{i+1} = (171\ X_i) \bmod 30269 \tag{2.3.1}$$

$$Y_{i+1} = (172\ Y_i) \bmod 30307 \tag{2.3.2}$$

$$Z_{i+1} = (170\ Z_i) \bmod 30323 \tag{2.3.3}$$

and then define

$$U_{i+1} = \{(\frac{X_{i+1}}{30269}) + (\frac{Y_{i+1}}{30307}) + (\frac{Z_{i+1}}{30323})\} \bmod 1 \tag{2.3.4}$$

Note that three "seeds" are required to start the generator. A seed can be any positive odd integer whose value is less than the applicable modulus.

One of the principle advantages of being able to generate random numbers arithmetically is the ability to produce the same sequence of random numbers whenever desired. Therefore, if one is comparing two alternative designs, then one is assured that the difference in the output measures of the experiment are due to differences in the alternative designs, not to experimental error.

Uniformly distributed random numbers in the interval [0,1] can be used to generate outcomes from any probability distribution. Taha[22] shows that by applying the method of inversion, where R is a [0,1] random number, the exponential distribution may be sampled by:

$$t = -\,MTBF\,\ln(R) \tag{2.3.5}$$

The Weibull distribution may be sampled by:

$$t = \eta\ (-\ln(R))^{1/\beta} \tag{2.3.6}$$

Unfortunately, the inversion method cannot be used with continuous distributions whose cumulative density function cannot be determined analytically. Typical examples are the normal, gamma and poisson distributions. Taha[22] states that for a pair of [0,1] random numbers R_1 and R_2, the random variable x defined as:

$$x = \sqrt{-2\ln R_1}\,\cos(2\pi R_2) \tag{2.3.7}$$

is standard normal with mean 0 and variance 1. Therefore, the normal distribution may be sampled by:

$$t = \mu + \sigma x \tag{2.3.8}$$

Note, if the value sampled from the normal distribution is negative then the simulator automatically sets the value to zero.

2.4 CAT COMPONENTS

The preceding discussion concerned the means of generating random failure times for single components. Often, one finds components which consist of subcomponents arranged in parallel. A failed subcomponent may not necessarily cause the component to fail, i.e. the component may have more than one life, hence the term cat component. This section discusses two common configurations which have been included in the simulator.

2.4.1 STANDBY CONFIGURATION

When N subcomponents are arranged in standby configuration, only one subcomponent can be active at a time. The cat component will therefore only fail once all the subcomponents have failed.

If the subcomponents are ordered in the sense that when the first one fails, the second one is switched into operation and when the second one fails the third one is switched into operation and so on, until the last subcomponent (N^{th} subcomponent) has failed. Then the time to fail for the cat component is simply the sum of each subcomponent lifetime t_i, i.e.

$$t_s = \sum_{i=1}^{N} t_i \qquad (2.4.1.1)$$

Note, each subcomponent lifetime is measured from the instant it is activated. For the simulator it was assumed that the components do not fail in the standby mode and that the switching mechanism is failure free.

The standby cat component is one of the more interesting to study by the simulation approach because of its great simplicity over the analytical method which can become quite difficult if the subcomponent failure densities are different from one another.

2.4.2 K OUT OF N CONFIGURATION

All subcomponents in this configuration are initially active and they remain so until they fail. The cat component requires a minimum number of subcomponents to be active in order to survive, e.g. 3 out of 5 (K out of N) subcomponents must be active.

If the subcomponent times to fail are ordered so that $t_1 \leq t_2 \leq .... t_n$, then the time to failure t_s for a K out of N cat component is:

$$t_s = t_{N-K+1} \qquad (2.4.2.1)$$

Note, for the K out of N configuration all subcomponents are initially active whereas for the standby configuration only one subcomponent can be active at a time.

2.5 SYSTEM FAILURE TIME

The previous two sections discussed means of generating random failure times for components and cat components from their respective probability distributions. In this section we examine how to determine the system failure time given the component failure times.

Forry[13] used a methodology (the search technique) of working from left to right through the system network, determining the minimum time to failure at each node of the reliability network. The minimum time to failure at a node would be determined by examining the time to failure of each input path into the node. The minimum time would then be the value of the input path with the smallest time to failure. The value at the final node would then be the system time to failure. The procedure was developed from the well known PERT method (Program Evaluation Review Technique).

Dubi[9] explained that the state of the system depends on its structure (the Reliability Block Diagram) and on the status of the components comprising the system. The function which determines the state of the system from the status of the components is called the system function (ISYSUP). The system function can only be one (operational) or zero (failed) and for the purposes of the simulator, the component status K can also only be one (operational) or zero (failed). The system will be up (ISYSUP=1) as long as their is a tie of operational components (K=1) from the entry point to the exit point of the system.

Consider the network system shown previously in **Figure 1.4.1**. There are four tie sets which may be listed as (1,3,5), (1,3,6), (2,4,6) and (2,4,7). The system function ISYSUP can therefore be constructed as:

$$IS = K(1)\times K(3)\times K(5)+K(1)\times K(3)\times K(6)+K(2)\times K(4)\times K(6)+K(2)\times K(4)\times K(7) \qquad (2.5.1)$$

where

$$ISYSUP = \{ \begin{matrix} 1 & If\ IS > 0 \\ 0 & Otherwise \end{matrix} \qquad (2.5.2)$$

The simulator generates a list of candidate system failure times by checking the value of ISYSUP at each stochastic event. The smallest value on this list will then be the system failure time.

The method used by Forry[13] requires the user to input a matrix of zeros and ones for each system. This is tedious and leads to many user input errors. The approach used by Dubi[9] requires the entering an equation into the program rather than a matrix. The only disadvantage of this is that the program requires compiling and linking for each system function. It was decided for practical reasons to adopt the approach used by Dubi[9] for the simulator.

2.6 SYSTEM AVAILABILITY

The determination of system availability is considerably more difficult than the determination of system reliability.

If the failure and repair times are exponentially distributed, application of Markov theory can produce solutions for a few systems. Dhillon[7] used Markov theory to derive the availability for a single component, i.e.

$$A_i(t) = \frac{\mu}{\lambda+\mu} + \frac{\mu}{\lambda+\mu} \exp[-(\lambda+\mu)t] \qquad (2.6.1)$$

Where λ and μ are the failure and repair rates respectively.

The time dependent term in the above equation decreases rapidly with time t and within a few cycles of operation, the system approaches the steady state availability which is independent of time, i.e.

$$A_i(t) = \frac{\mu}{\lambda+\mu} \qquad (2.6.2)$$

The Markov solution for a single component required the solving of two simultaneous differential equations. The same procedure could be used to find the availability of systems with large numbers of components. Unfortunately, the solution becomes impractical due to the large number of differential equations. If the component failure and repair densities are non-exponential then even Markov theory is no longer applicable. This results in an almost impossible situation to resolve in the analytic form. Fortunately, the Monte Carlo approach is quite simple with the accuracy of results being controlled by the cost of computer time.

It is important to note, before describing the manner in which the simulation model calculates availability, that at least two basic repair policies could be adopted. We could repair all failed components when the system fails, or we could repair components as they fail individually. The latter policy is rarely found in practice, although intuitively it may yield a higher system availability than the former policy. The simulator only repairs components following a system failure.

The manner in which the simulator calculates system availability is best described by considering the i^{th} and $(i+1)^{th}$ cycle of the program.

For the i^{th} cycle:

The system failure time t_s is determined from the component failure times. The component failure times are then searched to find those failure times which are equal to or less than t_s. This identifies which components have failed. Times to repair the failed components are then generated. The system repair time t_r is then calculated from the component repair times. The system failure time t_s and system repair time t_r are added to accumulators for system uptimes and downtimes.

For the $(i+1)^{th}$ cycle:

- Those components which had not failed by the end of system repair have an adjusted time to failure of $t_{i+1} = t_i-(t_s+t_r)$. Note, in this case the components were left on during system repair.

  If the components were switched off during system repair then they have an adjusted time to failure of $t_{i+1} = t_i-t_s$.

- Those components which failed at or before system failure have a completely new time to failure t_{i+1} generated.

- Those components which were left on during system repair and failed during system repair have their time to failure t_{i+1} set to zero.

The system failure time t_s and system repair time t_r are then calculated and added to the accumulators in the same way as for the i^{th} cycle.

At the end of the n^{th} system repair cycle an estimate of system availability $\hat{A}_s(t_n)$ is made by:

$$\hat{A}_s(t_n) = \frac{\sum_{j=1}^{n} t_s^j}{\sum_{j=1}^{n} t_s^j + \sum_{j=1}^{n} t_r^j} \quad , \; j=1,2\ldots,n \text{ cycles} \qquad (2.6.3)$$

Where t_n is the sum of the system failure and repair times at the end of the n^{th} system repair cycle.

2.7 DATA ANALYSIS

The final step in the Monte Carlo reliability and availability estimation process is the analysis of the simulator output. One must remember that the primary output data from the simulator is merely a sample of system times to failure and a sample of system times to repair.

A nonparametric or parametric approach can be taken to process the raw data from the simulator into reliability and availability estimates.

2.7.1 NONPARAMETRIC APPROACH

If the program user has no information regarding the underlying system time to failure distribution, he can make use of a nonparametric or distribution free method to obtain a point and confidence interval estimate of the system reliability and availability. This approach was programmed into the simulator.

2.7.1.1 RELIABILITY ESTIMATE

One can consider the sample of n system failure times generated by the model to be a random sample of the underlying distribution function F(t). The empirical cumulative time to failure distribution function $F_n(t)$ can then be defined as:

$$F_n(t) = \frac{j}{n} \ , \ t_j \le t < t_{j+1} \ , \ j = 1,2,\ldots,n-1 \qquad (2.7.1.1.1)$$

As the sample size increases, the deviation between $F_n(t)$ and F(t) tends toward zero and since reliability is defined as:

$$R(t) = 1-F(t) \qquad (2.7.1.1.2)$$

then the empirical reliability function can be defined as:

$$R_n(t) = 1-F_n(t) \qquad (2.7.1.1.3)$$

A simulation is a statistical experiment whose results are subject to experimental error. Hence, the setting of confidence intervals for point reliability estimates is important.

Forry[13] states that if one considers $F_n(t)$ to be the ratio of total failures to the total number of n trials at time t, then F(t) is the parameter q of the binomial distribution:

$$P[n \times F_n(t)=j] = \binom{n}{j} q^j (1-q)^{n-j} \qquad (2.7.1.1.4)$$

Where j is the number of failures and $F_n(t)$ is the maximum likelihood, minimum variance and unbiased estimate of q.

Likewise, R(t) is the parameter p of the binomial distribution:

$$P[n \times R_n(t)=k] = \binom{n}{k} p^k (1-p)^{n-k} \qquad (2.7.1.1.5)$$

Where k is the number of survivors at time t and $R_n(t)$ is the maximum likelihood, minimum variance and unbiased estimate of p.

Using this expression, a $(1-\alpha)100$ percent lower one sided confidence limit p^* can be determined by solving:

$$\sum_{y=k}^{n} \binom{n}{y} p^y (1-p)^{n-y} = \alpha \qquad (2.7.1.1.6)$$

for p, where

$$k = n \times R_n(t) \qquad (2.7.1.1.7)$$

Hines and Montgomery[16] state that the lower confidence limit for a one sided interval is chosen so that

$$P\{L \le \theta\} = 1-\alpha \qquad (2.7.1.1.8)$$

The interpretation of this is that there is a $100(1-\alpha)$ percent probability that the true θ is greater than L.

Note, the longer the confidence interval, the more confident we are that the interval actually contains the true value of θ. On the other hand, the longer the interval, the less information we have about the true value of θ. Ideally, one should obtain a relatively short interval with high confidence.

Unfortunately for large n, the computations required to find p from the binomial distribution become lengthy. Hines and Montgomery[15] state that for large n and binomial parameter p or $q < 0.1$, the binomial distribution is approximated by the poisson distribution with parameter np or nq. They also state that for np or $nq > 5$ the normal distribution with mean $\theta = np$, variance $\sigma^2 = np(1-p)$, and random variable $nR_n(t)$, gives a good approximation to the binomial distribution. Therefore, for np or $nq > 5$ the normal approximation can be used to determine the $(1-\alpha)100$ per cent lower confidence limit. For the region outside these limits, the poisson approximation can be used as long as $n > 50$ which is usually the case for the simulator.

The use of these two approximations simplifies the computations required to set the desired confidence levels for point estimates of reliability.

2.7.1.2 AVAILABILITY ESTIMATE

Point estimates $\hat{A}_s(t_n)$ for system availability $A_s(t_n)$ can be made from the sequence of n simulated system failure times (t_s^j) and repair times (t_r^j). This estimate was given in equation 2.6.3 as:

$$\hat{A}_s(t_n) = \frac{\sum_{j=1}^{n} t_s^j}{\sum_{j=1}^{n} t_s^j + \sum_{j=1}^{n} t_r^j}$$

Forry[13] explains that the sample variance σ_n^2 can be determined from the variances of the failure times V_s and repair times V_r by:

$$\sigma_n^2 = \frac{\left(\frac{V_s}{\bar{T}_s^2} + \frac{V_r}{\bar{T}_r^2}\right)}{16n} \qquad (2.7.1.2.1)$$

Where $\bar{T}_s$ and $\bar{T}_r$ are the sample means of the system failure and repair times respectively. Forry[13] also shows that the lower one sided confidence limit for availability is:

$$\frac{(1-Q)\hat{A}_s(t_n)}{Q(1-\hat{A}_s(t_n)) + (1-Q)\hat{A}_s(t_n)} \le A \le 1 \qquad (2.7.1.2.2)$$

where

$$Q = \hat{\sigma}_n K_{(1-\alpha)} + 0.5 \qquad (2.7.1.2.3)$$

and $K_{(1-\alpha)}$ is the $(1-\alpha)$ level of the standard normal distribution with mean of zero and variance of one.

2.7.2 PARAMETRIC APPROACH

This approach may give a more satisfactory reliability or maintainability estimate, if the engineer has some prior knowledge regarding the form of the system time to failure and repair distributions.

In this approach, estimates are made of the distribution parameters from the sample of n system failure and repair times. From these, point estimates of system reliability and maintainability can be made. The chi-square statistic can be used to test the hypothesis that the observed failure or repair times are from the assumed density.

The chi-square statistic is calculated from the sample by:

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \qquad (2.7.2.1)$$

Where K is the number of class intervals. The quantity O_i is the observed frequency in the i^{th} class interval. The expected frequency in the i^{th} class interval from the hypothesized probability distribution is denoted by E_i. The test is made by comparing χ^2 (chi-square) computed from the sample, with an α sized critical value of the chi-square distribution with k-m-1 degrees of freedom. Where the quantity k is the number of class intervals and m is the number of parameters estimated from the sample. Hines and Montgomery[*] state that the hypothesis would be rejected if:

$$\chi^2 \geq \chi^2_{\alpha,k-m-1} \qquad (2.7.2.2)$$

where

$$\chi^2_{\alpha,k-m-1} = c \qquad (2.7.2.3)$$

is the solution to

$$\int_c^{\infty} f(x)\; dx = 1-\alpha \qquad (2.7.2.4)$$

and f(x) is the chi-square density with k-m-1 degrees of freedom.

Commercially available statistical packages such as **STATGRAPHICS®** offer extensive distribution fitting facilities which include the chi-square test. Therefore, it was decided not to program distribution fitting facilities into the simulator. Rather, files containing the sample failure and repair times are made available for exporting into any statistical package.

2.8 EMBEDDED LOGIC

It is imperative for the user to understand the embedded logic of the code. This will help with the interpretation of the modelling results and also in determining the applicability of the simulator to solving the problem.

- The system function is limited to network systems only. Network systems are those systems whose logic can be fully represented by a reliability block diagram.

- All component failures are repaired following a system failure.

- The system repair time is based on the repair times of all the components which have failed. The system repair time may be the average component repair time, the worst component repair time or the sum of each component repair time.

- If a component fails before or at the system failure time, then the simulator generates a completely new time to failure for the component.

- If a component has not failed by the system failure time and remains active during system repair, and has still not failed by the end of system repair, then the time to failure of the component is reduced by the system failure time and the system repair time.

- If a component remained active during system repair and failed while the system was being repaired or immediately when the system was repaired, then the time to failure of the component is set to zero. Therefore, it is possible that the system could fail immediately when activated again.

- If a component has not failed before or at the system failure time and was switched off during system repair, then the time to failure of the component is reduced by the system failure time.

- The simulator calculates the average availability over the whole time interval starting from time zero until the required number of histories have been completed.

- The time to repair density for a cat component applies to the component as a whole and not to the individual subcomponents, whereas the failure density applies to individual subcomponents.

- If a reliability/availability run is selected, the empirical reliability distribution is based upon system times to fail from the last repair. This data therefore estimates reliability as a function of the maintenance option and is not necessarily representative of the non-maintained system reliability. Of course, the non-maintained system reliability is estimated by running the pure reliability option.

3 PROGRAM DESCRIPTION

Microsoft FORTRAN 4.1* was used to program the simulator. In order to use the simulator, a FORTRAN compiler must be available to program the system function. The simulator should work with most FORTRAN compilers and will run on most computers, even small personal computers where the computer word size is small.

The program consists of a main program and several subroutines. The main program reads in general run control data and component information. The main program then calls into operation the necessary subroutines for processing the input data.

Using the above information, the program generates a listing of the system times to fail, a table containing the failure time histogram, and the estimated point reliability. If the user specifies that a reliability/availability simulation is to be performed then an estimate of the average availability is given in addition to the empirical reliability distribution. A listing of the system times to repair is also generated. It should be emphasized that the simulator determines an average steady state availability and not the availability at specific points in time.

The user can specify whether components are switched off or left on during system repair. The user can also select between different options for determining system repair time, i.e. the maximum component repair time, the sum of the component repair times or the average component repair time.

If the user has some knowledge of the underlying reliability or maintainability distributions, then the listing of system failure or repair times can be exported to a statistical package where the distribution parameters can be determined. The chi-square goodness of fit test may be used to test the assumed distribution. Reliability or maintainability predictions can then be made using the hypothesized distribution.

The program listing for the simulation model can be found in **Appendix A.** The reader is advised to refer to the relevant program listing while reading the sections which follow.

3.1 INPUT DESCRIPTION

The data input for the simulator consists of a data file called **RAMIN** and the system function which is programmed into subroutine **SYST**. The data file contains general run control data and component information. The system function contains the reliability network information.

The variables in **RAMIN** which contain the general run control data are:

NTYPE	-	An integer indicating the desired run type. NTYPE = 0, Reliability simulation only NTYPE = 1, Reliability/Availability simulation with components switched off during repair NTYPE = 2, Same as 1 but with components left on during repair
NTIME	-	An integer indicating the number of failure and/or repair cycles to go through (sample size), limited to 5000.
N	-	An integer indicating the number of components in the system, limited to 20.
KFIX	-	An integer indicating the type of repair time to be used. KFIX = 1, Sum of component repair times KFIX = 2, Maximum component repair time KFIX = 3, Average component repair time
IPROB	-	An integer for the user's run number identification
ISZE	-	An integer indicating the number of class intervals to be used in the simulated time distribution table.
FI	-	A real number indicating the class interval width for the simulated time distribution table.

The variables in **RAMIN** which contain component data are

ICODE	-	An integer indicating whether a component is a subsystem of N parallel (K out of N) or N standby identical components, i.e. a cat component. ICODE = 0, Normal component ICODE = NK, N identical components in parallel with K out of N required for success. ICODE = -N, N identical components in standby configuration

KFDN(I)	-	An integer indicating the type of component failure density for component I. KFDN(I) = 1, Exponential KFDN(I) = 2, Normal KFDN(I) = 3, Weibull KFDN(I) = 4, Lognormal
KRDN(I)	-	The same as above except that the integer indicates the component repair density.
FPTR(I,1)	-	A real number indicating one parameter of the failure density for component I. Exponential - mean Normal - mean Lognormal - mean Weibull - characteristic life
RPTR(I,1)	-	Same as above except that the real number indicates one parameter of the repair density.
FPTR(I,2)	-	A real number indicating the second parameter of the failure density, if required, for component I. Normal - standard deviation Lognormal - standard deviation Weibull - shape factor
RPTR(I,2)	-	Same as above except that the real number indicates the second parameter of the repair density, if required.

The system function ISYSUP is entered by the user into subroutine **SYST** below the block entitled:

```
*******************************************
* ENTER THE SYSTEM FUNCTION HERE *
*******************************************
```

Subroutine **SYST** must then be compiled and linked with the all the other program object files to obtain the executable file. Examples of **RAMIN** and system functions can be found in the examples which were prepared for this report.

3.2 PROGRAM MAIN

Program execution begins with reading data from **RAMIN** and writing this data to the main output file **RAMOUT**. The program then goes on to complete NTIME histories.

For each history of a pure reliability simulation, the program calls subroutine **FAILT** which in turn calls subroutines **ETIME**, **RAND**, **PARL** and **STBY** as required. Subroutine **FAILT** returns a random time to fail for each component of the system. The program then calls subroutine **SYST** which returns a system time to fail, using the previously generated component failure times. In this mode, the program skips over the code which is used to determine the system repair times and availability. Each system failure time is stored in the vector TSYSF which is written to a file called **TTFLIST**. A pure reliability simulation is specified by setting NTYPE to zero. MSW is a program control variable which controls the call to subroutine **FAILT** which depends on whether a pure reliability or a reliability/availability simulation is required.

For each history of a reliability/availability simulation the program calls subroutine **SYST** which returns a system time to failure. The program then checks which components have failed at or before the system failure time. The program then calls subroutine **ETIME** which in turn calls subroutine **RAND** and returns a random time to repair for each failed component. The program then determines a system time to repair from the individual component repair times. The system repair times are stored in the vector TSYSR and are written to a file called **TTRLIST**. The system repair time is calculated according to the KFIX specification. The NTYPE specification determines whether components are switched off or left on during system repair. The program then proceeds to accumulate information such as the sum of system repair and failure times for each history.

After all the histories have been completed, the program calls subroutine **TAB** NTIMES which calculates and prints the empirical reliability and availability statistics.

3.3 SUBROUTINE SYST

The purpose of this subroutine is to determine a system time to failure SYSF from previously determined component failure times. The subroutine is called by program **MAIN** to which it returns the parameter SYSF.

The component failure times are stored in vector T. KST is a vector containing the status of each component. TEMP2 is a temporary storage vector which facilitates the determination of the system failure time. A record of the number of failures per component NF is also kept as the simulation proceeds.

The system function ISYSUP is entered into the subroutine below the block entitled:

```
*******************************************
* ENTER THE SYSTEM FUNCTION HERE *
*******************************************
```

The user is then required to compile the subroutine after entering the system function. This object file must then be linked together with all the other object files to obtain the executable file.

This subroutine must return a system failure time to program **MAIN** each time it is called. A check was therefore built in which aborts program execution if subroutine **SYST** is unable to return a system failure time. One should not encounter this problem with network type systems.

3.4 SUBROUTINE FAILT

The purpose of this subroutine is to determine the time to fail TI for a cat component. A cat component consists of a subsystem of components either in an active parallel or standby configuration.

This subroutine is called by program **MAIN** to which it returns the parameter TI for a single component or a cat component. The subroutine calls subroutine **ETIME** once for a single component and a number of times for a cat component. Function **STBY** and function **PARL** are also used to determine the random time to fail for cat components.

KF is the identification code for the component failure density. FI represents the mean for the exponential, normal and log normal distributions and the characteristic life for the Weibull distribution. FJ represents the standard deviation for the normal and log normal distributions and the shape parameter for the Weibull density. IC allows one to distinguish whether the item is a single component, a K out of N subsystem or a standby subsystem.

3.5 FUNCTION STBY

This function computes a random time to fail for a subsystem of NEL components in a standby configuration. The component times to fail are determined previously and stored in the vector T prior to entering the subprogram. The function is called by subroutine **FAILT**.

3.6 FUNCTION PARL

This function finds the time to fail for an NSUS out of NEL components (i.e. a K out of N arrangement) in active parallel redundancy whose random times to fail have been previously determined and stored in the vector PT. The function is called by subroutine **FAILT**.

3.7 SUBROUTINE ETIME

This subroutine allows one to sample a random time to fail or repair from the negative exponential distribution with MTBF P1, the normal and lognormal distributions with mean P1 and standard deviation P2, and the Weibull distribution with characteristic life P1 and shape parameter P2.

This program is called by subroutine **FAILT** to which it returns a random time to fail. The program is also called by program **MAIN** to which it returns a random time to repair. Random numbers to facilitate the sampling from each distribution are returned to this program by subroutine **RAND**.

3.8 SUBROUTINE RAND

The function of the subroutine is to generate uniformly distributed random numbers in the interval [0,1]. The program is called by subroutine **ETIME**.

A combination congruential generator was used to generate random numbers. This particular generator requires the output from three separate congruential generators. This subroutine therefore calls subroutines **RAND1**, **RAND2** and **RAND3** for these outputs.

3.9 SUBROUTINE TAB

The purpose of this subroutine is to produce the empirical reliability and availability output blocks. The availability output block is only produced for a reliability/availability simulation. This program is called by program **MAIN**.

As the previously determined variable A is passed to subroutine **TAB**, the count of entries in the interval of KFREQ is augmented by one. The number of intervals ISZE and the width of the intervals FI are specified by the user. Information to subsequently compute the mean and variance of the sample of NT system failure times is updated. MS controls whether or not the present call of subroutine **TAB** is the first or a later call. On the first call, certain accumulators and other variables are initialised.

When NT calls of **TAB** have been executed, the mean and variance ~~of the~~ sample system failure and repair times are determined. The average availability with standard deviation and estimates of the lower 90 % and 95 % confidence levels are then computed. Finally, the empirical reliability function is computed along with lower 95 % confidence level estimates.

3.10 OUTPUT DESCRIPTION

Three reports are generated by the model, i.e. a general output file called **RAMOUT**, a file **TTFLIST** which contains the list of system fail times and a file **TTRLIST** which contains the list of system repair times.

RAMOUT comprises of an input data block, an availability block, a reliability block and a failure block.

The input block displays the data which was input by the user. It is always good practice to check the input data which has been read by the program.

The availability block is only generated when the simulator operates under the reliability/availability option. The average system uptime and the average system downtime are displayed together with standard deviations for each value. The average steady state availability is then displayed together with lower 90 % and 95 % confidence level estimates.

The reliability block is generated for all runs. If the reliability/availability option is used, then the empirical reliability distribution is based upon system times to fail from the last repair. This data estimates reliability as a function of the maintenance option and is not necessarily representative of the non-maintained system reliability.

The reliability block displays the average system time to fail and the standard deviation. Thereafter, the system time to failure distribution table is shown. Column 1 is the time at the end of each class interval. Column 2 is the number of failures occurring in the interval. Column 3 is the lower 95 % confidence level estimate of system reliability (R95L). Column 4 is the maximum likelihood estimate of system reliability (RMLE). At the end of the table an indication is given of the outliers which occurred because of the chosen number of class intervals and the width of class intervals. Finally the number of class intervals, the class interval width, the maximum system failure time and the minimum system failure time are displayed.

The failure block is generated for all runs. This block lists the cumulative number of component failures per component for all histories.

The file **TTFLIST** is generated for all runs and it contains a list of simulated system failure times. The file **TTRLIST** is generated only for the reliability/availability run and it contains a list of simulated system repair times. If the user has some knowledge of the form of the distributions then these files may be exported to a statistical analysis package such as **STATGRAPHICS®** where the distribution parameters can be determined.

4 MODEL VALIDATION AND APPLICATION

Examples of several systems were processed by the simulator to validate the model and demonstrate various applications of the model. Model validation consisted of comparing the reliability and availability estimates produced by the simulator with analytically determined values where feasible. **SPAR®** and **AMIR®** were used to validate the simulator estimates where this was not feasible.

The examples were extracted from various references in the aerospace and industrial environments. Empirical reliability and/or average availability predictions were accomplished for each system. Various simulator options were demonstrated, i.e. the different ways of calculating the system repair time as well as leaving components on or switching them off during system repair. Histograms of system times to fail or repair were also displayed for some systems.

One is usually interested in the reliability of a system at a certain time. However, for some systems the chances of system failures occurring during this time are extremely remote. For example, one may wish to estimate the reliability of a Boeing 747 electrical system for a typical flight time of 3 hours. But the chances of a catastrophic failure during this time are extremely remote. In order for the simulator to see such an event during this time requires millions of histories which becomes impractical due to computer time limitations. This problem can be overcome by using biasing techniques such as those used by Goldfeld and Dubi[14]. These biasing techniques were not included in this study. Therefore, all the reliability results computed by the simulator are grouped at reasonable time intervals around the mean life.

4.1 HELICOPTER ELECTRONIC WARFARE SYSTEM

This example illustrat[illegible] use of the simulator in predicting the mission reliability of an Electr[illegible] Warfare System (EWS) for a combat helicopter. The system configuration consists of a series-parallel arrangement of components with a region of K out of N active redundancy. As the system comprises of electronic components, negative exponential failure densities were selected throughout. It was also possible to validate the simulator results analytically.

The main function of the EWS is to make the aircrew aware of the existence, position and direction of any hostile radar during a mission. The mission reliability model (refer **Figure 4.1.1**) is based on the assumption that the warning against the existence of threats is mission critical but not the position or direction thereof.

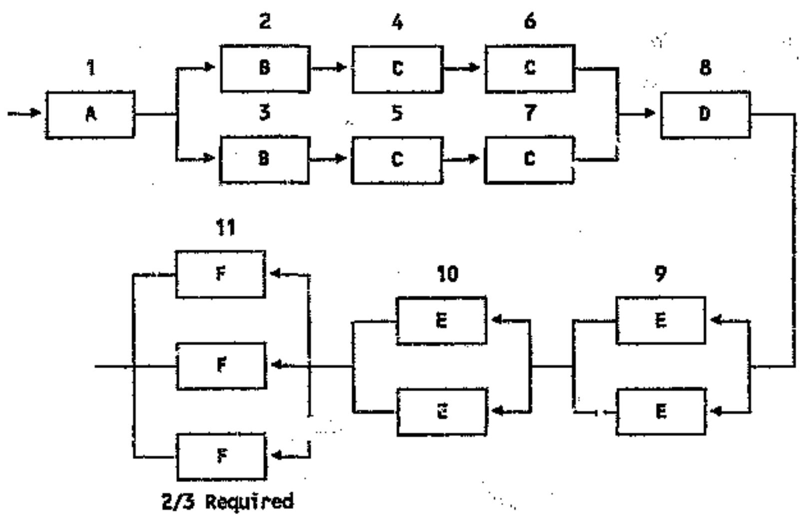

Figure 4.1.1 Electronic Warfare System - RBD

The logic used to construct the RBD from the physical system according to the mission reliability requirement can be described as follows:

- The EW controller$^{(A)}$ is the crux of the EW system. When this component fails all the EW functions are lost which makes the controller a mission critical item.

- The two dual front end recorders$^{(B)}$ are each connected to two radar warning antennas$^{(C)}$. The assumption here is that in the case of one dual front end recorder failing, two of the four sensing functions are lost, one on each side of the helicopter. In this case the threat direction indication will be degraded. Detection will however still be possible and the mission will not be aborted. If both dual front end recorders fail all the laser sensing functions are lost which will cause a mission abort. Two radar warning antennas may fail as long as they are not situated on the same side of the helicopter.

- The laser warning analyzer(D) receives the signals from the laser detectors and calculates the direction of the source. In the case of a failure of the laser warning analyzer, the function of detection is lost which results in a mission abort. The failure of two laser detectors(E) will not create a mission abort situation as long as they are not situated on the same side of the helicopter. The hostile fire sensor units(F) may be considered as a 2 out of 3 system for mission purposes.

The failure properties of the above components are listed in **Table 4.1.1.**

Table 4.1.1 EWS Component Failure Data

Component	Description	MTBF (hrs)
A	Electronic Warfare Controller	800
B	Dual Front End Recorder	2500
C	Radar Warning Antenna	10000
D	Laser Warning Antenna	1000
E	Laser Detector	344
F	Hostile Fire Sensor Unit	619

The system function **ISYSUP** is easily deduced from the tie sets contained in **Figure 4.1.1** and is:

$$ISYSUP = K1 + K2 \qquad (4.1.1)$$

where

$$K1 = KST(1) \times KST(3) \times KST(5) \times KST(7) \times KST(8) \times KST(9) \times KST(10) \times KST(11) \qquad (4.1.2)$$

$$K1 = KST(1) \times KST(2) \times KST(4) \times KST(6) \times KST(8) \times KST(9) \times KST(10) \times KST(11) \qquad (4.1.3)$$

Note, components 9, 10 and 11 each consist of more than one subcomponent, i.e. they are cat components. This considerably simplified the programming of the system function.

The system function **ISYSUP** was entered into subroutine **SYST** (refer **Appendix B**) which was then compiled and linked to the other program files to form the executable simulator file. The component and general program control data were then entered into the input file **RAMIN** (refer **Appendix B**). A pure reliability simulation was selected (NTYPE = 0) with a sample size of 5000 (NTIME = 5000). The empirical reliability distribution was defined as having 12 class intervals (ISZE = 12) each of width 50 hours (FI = 50). Components 9 and 10 were entered as 1 out of 2 cat components in active redundancy (ICODE = 21). Component 11 was entered as a 2 out of 3 cat component in active redundancy (ICODE = 32). All other components were entered as single components (ICODE = 0). Negative exponential failure distributions were assigned to all components (KFDN(I) = 1).

The simulator output **RAMOUT** can be found in **Appendix B.** The results indicated a mean life of 169.05 hours and a standard deviation of 127.20 hours. The empirical reliability results have been redisplayed in **Table 4.1.2.** The column R95L indicates the lower 95 % confidence limit of reliability and the column RMLE the maximum likelihood estimate. For example, one can say with 95 % confidence that the mission reliability is greater than 0.835 at 50 hours while the most likely reliability is 0.843. In other words, the probability of the system surviving 50 hours is 0.843 and the probability of the system failing within 50 hours is 0.157.

The list of system times to fail **TTFLIST** was used to generate a histogram of system failure times (refer **Figure 4.1.2**). The histogram indicates a skewed distribution.

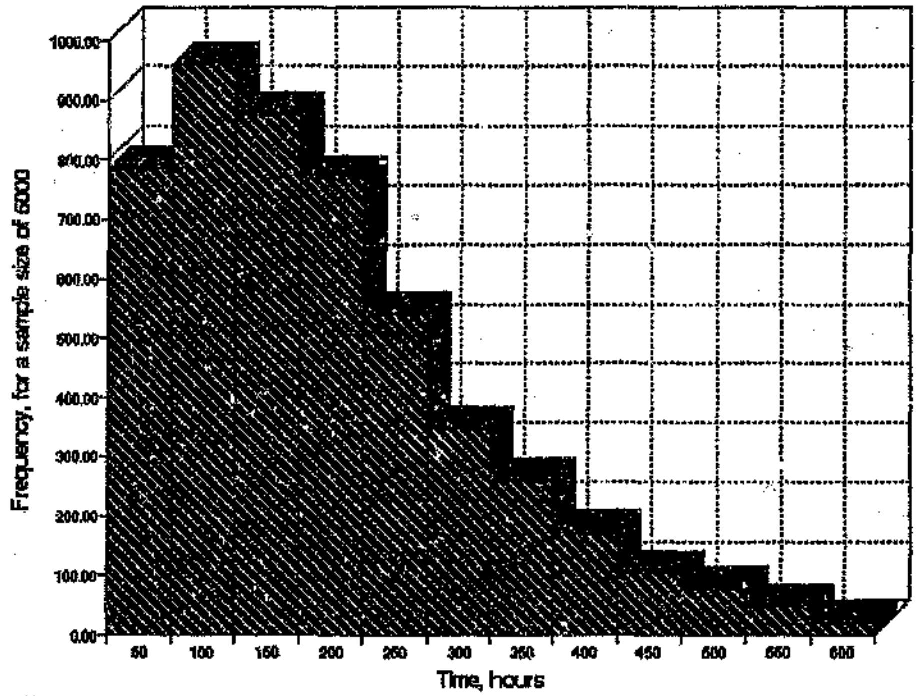

Figure 4.1.2 Histogram of System Failure Times - EWS

The true reliability function was derived to validate the simulator output and is given as:

$$R_s(t) = R_A(t) \times R_{12}(t) \times R_D(t) \times R_9(t) \times R_{10}(t) \times R_{11}(t) \qquad (4.1.4)$$

where

$$R_{12}(t) = 2R_B(t)R_C(t)^2 - R_B(t)^2R_C(t)^4 \qquad (4.1.5)$$

$$R_9(t) = R_{10}(t) = 2R_E(t) - R_E(t)^2 \qquad (4.1.6)$$

$$R_{11}(t) = R_F(t)^3 + 3R_F(t)^3(1-R_F(t)) \qquad (4.1.7)$$

The individual component reliabilities at time t are easily determined from **Equation 2.2.2.**

The results of computing the true reliability at times of [illegible] through to 600 hours are shown in **Table 4.1.2** along with the corresponding reliability estimate produced by the simulator. The simulator results compared favourably with the true results.

Table 4.1.2 Simulated vs True Reliability - [illegible]WS

Hours	R95L	RMLE	R True
50	0.835	0.843	0.846
100	0.640	0.651	0.656
150	0.464	0.476	0.478
200	0.311	0.322	0.332
250	0.204	0.214	0.223
300	0.136	0.144	0.145
350	0.086	0.093	0.093
400	0.053	0.058	0.058
450	0.033	0.038	0.036
500	0.019	0.022	0.022
550	0.011	0.013	0.013
600	0.007	0.009	0.008

4.2 SPACE SHUTTLE COMPUTER SYSTEM

This example illustrates the use of the simulator in selecting a computer system from seven alternative designs. The selection criteria were defined in terms of mission reliability and mean life. The example was extracted from Forry[13].

The different designs under consideration included series-parallel arrangements of components with regions of standby redundancy and K out of N active redundancy.

The alternative configurations were as follows:

(1) Central Simplex Computer System
(2) Central Dual Computer System
(3) Triple Processor Computer System A
(4) Triple Processor Computer System B
(5) Multi Processor Computer System A
(6) Multi Processor Computer System B
(7) Multi Processor Computer System C

The simulation results for the first three configurations were validated analytically.

The components making up the different designs were power supplies, input/output units, memory units, and central processor units. The failure properties of each component are shown in **Table 4.2.1**. Component failure distributions included exponential and normal distributions.

Table 4.2.1 Space Shuttle Computer System Component Data

Component	Description	Time to Failure (months)		
		Distr	Param 1	Param 2
A	Power Supply	Exp	7.5	N/A
B	Input/Output Unit	Normal	6.0	1.5
C	Central Processor Unit	Exp	10.0	N/A
D	Memory Unit	Exp	8.4	N/A

For each configuration, the system function **ISYSUP** was programmed into subroutine **SYST** and the component and general program control data were entered into **RAMIN.** A sample size of 5000 was selected for all configurations. Subroutine **SYST, RAMIN** and **RAMOUT** for each configuration can be found in **Appendix C.**

The design requirements were that the computer system must have a mean lifetime of at least 2.5 months, at least an 85 percent chance of surviving a one month operation, and at least a 25 percent chance of surviving a four month operation. The system reliability at one and four months and the mean life estimate for each configuration as obtained by the simulator are shown in **Table 4.2.2.**

Table 4.2.2 Estimated System Reliability and Mean Life - Alternative Shuttle Computer Systems

Config	R_s (1 month)	R_s (4 months)	Mean Life	Sigma
1	0.699	0.226	2.44	1.94
2	0.949	0.534	4.25	2.12
3	0.762	0.258	2.65	1.90
4	0.784	0.356	3.83	3.60
5	0.626	0.137	1.95	1.66
6	0.626	0.141	1.98	1.72
7	0.766	0.310	2.95	2.13

These results indicated that the Central Dual Computer System is the only configuration which could not be rejected as a candidate to meet the system mean lifetime and reliability specifications. All of the other systems would be rejected since at least one of their mean lifetime or reliabilities is below the requirement. Each configuration is discussed in greater detail in the pages which follow.

CONFIGURATION 1 – CENTRAL SIMPLEX COMPUTER SYSTEM

The Central Simplex Computer System is a simple series combination of one of each component. The failure of any one or more components results in the system being down. This logic can be represented by a simple series network (refer **Figure 4.2.1**). This design would not be expected to meet the reliability requirements but is useful for comparative purposes.

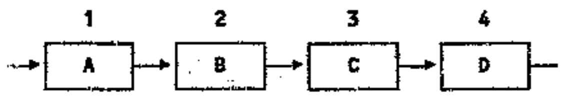

Figure 4.2.1 Central Simplex Computer System – RBD

The system function **ISYSUP** is defined as:

$$ISYSUP = KST(1) \times KST(2) \times KST(3) \times KST(4) \qquad (4.2.1)$$

The simulator results indicated a mean lifetime of 2.44 months and an estimated reliability of 0.699 at one month and 0.226 at 4 months. None of these values meet the specification.

The true reliability function is:

$$R_s(t) = e^{-0.352t} \int_t^{\infty} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6}{1.5}\right)^2} dx \qquad (4.2.2)$$

The results of this computation for times of one through eight months are shown in **Table 4.2.3** along with the estimates produced by the simulator. The reliability simulator produced estimates very close to the analytically derived values.

Table 4.2.3 Simulated vs True Reliability - Configuration 1

Months	R95L	RMLE	R True
1	0.689	0.699	0.703
2	0.485	0.497	0.493
3	0.331	0.342	0.340
4	0.216	0.226	0.222
5	0.122	0.130	0.129
6	0.052	0.057	0.060
7	0.015	0.018	0.021
8	0.003	0.004	0.005

CONFIGURATION 2 – CENTRAL DUAL COMPUTER SYSTEM

The Central Dual Computer System consists of a series arrangement of two parallel combinations of a memory unit in series with a central processor, two parallel input/output units, and two parallel power supplies. The reliability network is shown in **Figure 4.2.2**. The mean lifetime of this system should be greater than for the Central Simplex Computer System because of the increased redundancy.

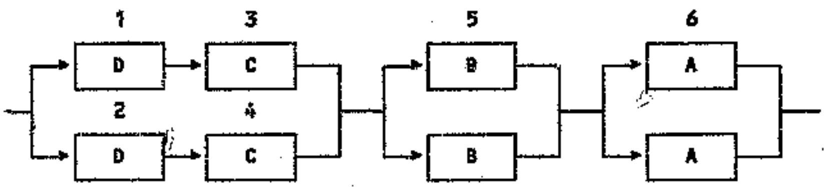

Figure 4.2.2 Central Dual Computer System – RBD

The system function **ISYSUP** is:

$$ISYSUP = KST(1) \times KST(3) \times KST(5) \times KST(6) + KST(2) \times KST(4) \times KST(5) \times KST(6) \quad (4.2.3)$$

Note, components 5 and 6 are cat components.

The simulator results indicated a mean lifetime of 4.25 months and an estimated reliability of 0.949 at one month and 0.534 at 4 months. The estimated mean lifetime is 1.74 times greater than the mean lifetime of the Central Simplex Computer System. The reliability estimates are also greater than those for the Central Simplex Computer System. In addition, all the reliability and mean lifetime specifications are exceeded.

The true reliability function is:

$$R_s(t) = R_1(t) \times R_2(t) \times R_3(t) \tag{4.2.4}$$

where

$$R_1(t) = 2e^{-.219t} - e^{-.438t} \tag{4.2.5}$$

$$R_2(t) = 2\int_t^{\infty} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6}{1.5}\right)^2} dx - \left[\int_t^{\infty} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6}{1.5}\right)^2} dx\right]^2 \tag{4.2.6}$$

$$R_3(t) = 2e^{-.133t} - e^{-.267t} \tag{4.2.7}$$

The results of computing the true reliability at times of one through ten months are shown in **Table 4.2.4** along with the estimates produced by the simulator. The reliability simulator produced estimates very close to the analytically derived values.

Table 4.2.4 Simulated vs True Reliability – Configuration 2

Months	R95L	RMLE	R True
1	0.944	0.949	0.946
2	0.513	0.822	0.855
3	0.662	0.673	0.682
4	0.523	0.534	0.548
5	0.380	0.391	0.399
6	0.228	0.237	0.244
7	0.098	0.106	0.107
8	0.029	0.034	0.032
9	0.004	0.006	0.005
10	0.000	0.001	0.001

CONFIGURATION 3 – TRIPLE PROCESSOR COMPUTER SYSTEM A

The reliability block diagram of the Triple Processor Computer System A is shown in **Figure 4.2.3**. In this design, two out of three central processor units and input/output units are required for successful operation. This redundancy in both the central processor and input/output units should provide a higher system reliability than that of the Central Simplex Computer System, but a lower system reliability than that of the Central Dual Computer System.

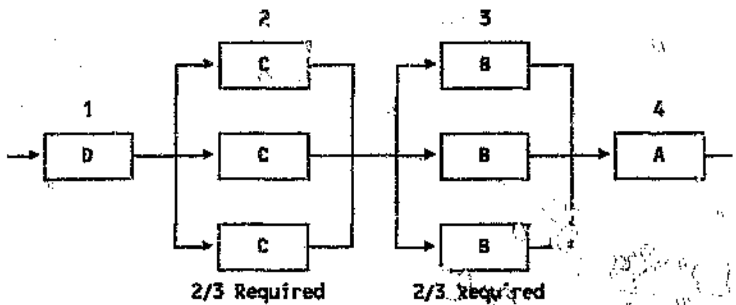

Figure 4.2.3 Triple Processor Computer System A – RBD

The system function **ISYSUP** is:

$$ISYSUP = KST(1) \times KST(2) \times KST(3) \times KST(4) \tag{4.2.8}$$

Note, components 2 and 3 are cat components.

The simulator results indicated a mean lifetime of 2.65 months and an estimated reliability of 0.762 at one month and 0.258 at 4 months. This shows some improvement in mean lifetime and reliability over the Central Simplex Computer System, but the reliability at one month still does not meet the specification.

The true reliability function is:

$$R_s(t) = R_1(t) \times R_2(t) \qquad (4.2.9)$$

where

$$R_1(t) = 3e^{-.452t} - 2e^{-.552t} \qquad (4.2.10)$$

$$R_2(t) = 3\left[\int_t^{\infty} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6}{1.5}\right)^2} dx\right]^2 - 2\left[\int_t^{\infty} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-6}{1.5}\right)^2} dx\right]^3 \qquad (4.2.11)$$

The results of computing the true reliability at times of one through eight months are shown in **Table 4.2.4** along with the corresponding reliability estimate produced by the simulator. The simulator results compare favourably with the true results.

Table 4.2.4 Simulated vs True Reliability - Configuration 3

Months	R95L	RMLE	R True
1	0.752	0.762	0.757
2	0.542	0.553	0.552
3	0.370	0.382	0.390
4	0.248	0.258	0.266
5	0.140	0.148	0.156
6	0.055	0.060	0.063
7	0.010	0.012	0.013
8	0.001	0.002	0.005

CONFIGURATION 4 – TRIPLE PROCESSOR COMPUTER SYSTEM B

The Triple Processor Computer System B is identical to System A configuration except the central processor units and input/output units are changed to a passive standby configuration of three units. This was accomplished by changing the input variable ICODE from 32 to -3.

The replacement of standby redundant units for active redundant units should increase the mean lifetime and reliability over those of the Triple Processor Computer System A. The results from the simulator confirmed this. The estimated mean lifetime of 3.83 months meets the requirement, but the estimated reliability of 0.784 at one month still does not meet the design requirement.

CONFIGURATION 5 – MULTI PROCESSOR COMPUTER SYSTEM A

This configuration consists of a Central Simplex Computer System combined in series with a Triple Processor Computer System A in which the 2 out of 3 active redundant central processor units and input/output units are changed to a 1 out of 3 active redundant configuration (ICODE = 31). The reliability network is shown in **Figure 4.2.4.**

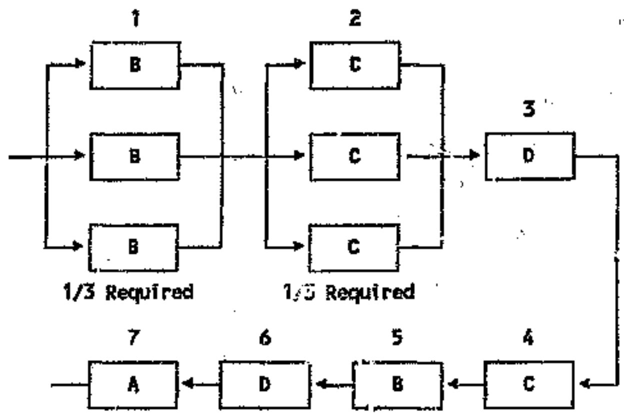

Figure 4.2.4 Multi Processor Computer System A – RBD

The system function **ISYSUP** is defined as:

$$ISYSUP = KST(1) \times KST(3) \times KST(5) \times KST(6) + KST(2) \times KST(4) \times KST(5) \times KST(6) \times KST(7) \qquad (4.2.12)$$

Note, components 1 and 2 are cat components.

Even though the mean lifetime and reliability of the Triple Processor Computer System A would have been increased by the change in redundancy, the series arrangement should cause the reliability and mean lifetime of this configuration to be less than that of the Central Simplex Computer System. The results from the simulator confirmed this, i.e. a mean lifetime of 1.95 months and estimated reliabilities of 0.626 and 0.137 at one and four months respectively.

CONFIGURATION 6 – MULTIPROCESSOR COMPUTER SYSTEM B

The Multiprocessor Computer System B is the same as the Multiprocessor Computer System A except the active redundant central processor units and input/output units are changed to standby redundant configuration, i.e. ICODE for components 1 and 2 were changed from 31 to −3.

The results show a very slight improvement over the Multiprocessor Computer System A, i.e. a mean lifetime of 1.98 months and estimated reliabilities of 0.626 and 0.141 at one and four months respectively. The reliability at one month remained the same.

CONFIGURATION 7 - MULTI PROCESSOR COMPUTER SYSTEM C

This configuration consists of a serial arrangement of 2 out 4 active redundant configurations of central processor units and input/output units, a power supply, and a memory unit. The reliability network is shown in **Figure 4.2.5.**

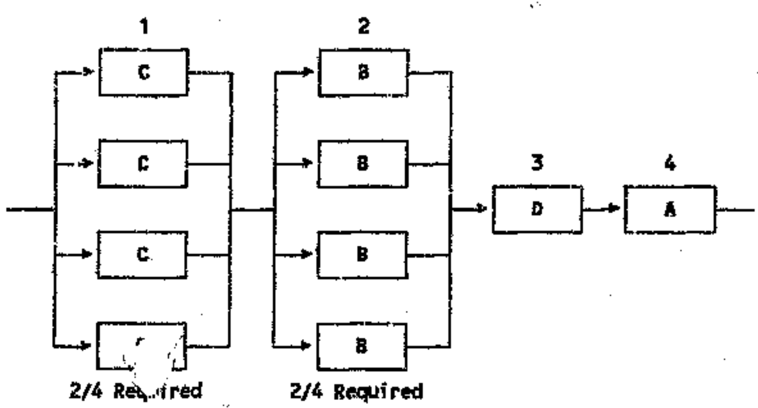

Figure 4.2.5 Multi Processor Computer System B - RBD

The system function **ISYSUP** is defined as:

$$ISYSUP = KST(1) \times KST(2) \times KST(3) \times KST(4) \qquad (4.2.13)$$

Note, components 1 and 2 are cat components.

As expected the simulator results indicated that the greater redundancy results in a longer mean lifetime and estimated reliability than those of the previous two multiprocessors. The estimated mean lifetime was 2.95 months with estimated reliabilities of 0.766 and 0.310 at one and four months respectively.

4.3 BOEING 747 ELECTRICAL POWER SYSTEM

This example uses the simulator to estimate the reliability of a 115V AC power bus which forms part of the Boeing 747 electrical power system. This example was extracted from a Boeing reliability engineering report compiled by Barry[1] in 1969. As the report is relatively old, the configuration of the current electrical power system may be considerably different to that presented in this analysis. Barry[1] also noted that the study was preliminary and would be revised upon receipt of more detailed information. Exponential failure distributions were assumed for all components.

The reliability logic of the electrical power system is not easily solved by analytic means. One therefore had to resort to a computer model such as the simulator to solve the problem. The results obtained by the simulator were validated using **AMIR***.

Aircraft are often dispatched with systems which are not 100 % operational. In this example, the 115V AC system was analyzed for the case where it is 100 % operational at dispatch. The reliability block diagram for the system is shown in **Figure 4.3.1.**

The physical power system consists of a large number of components. Fortunately, the reliability logic of the system allows one to lump together many components in series. This simplifies the simulation model and reduces computation time.

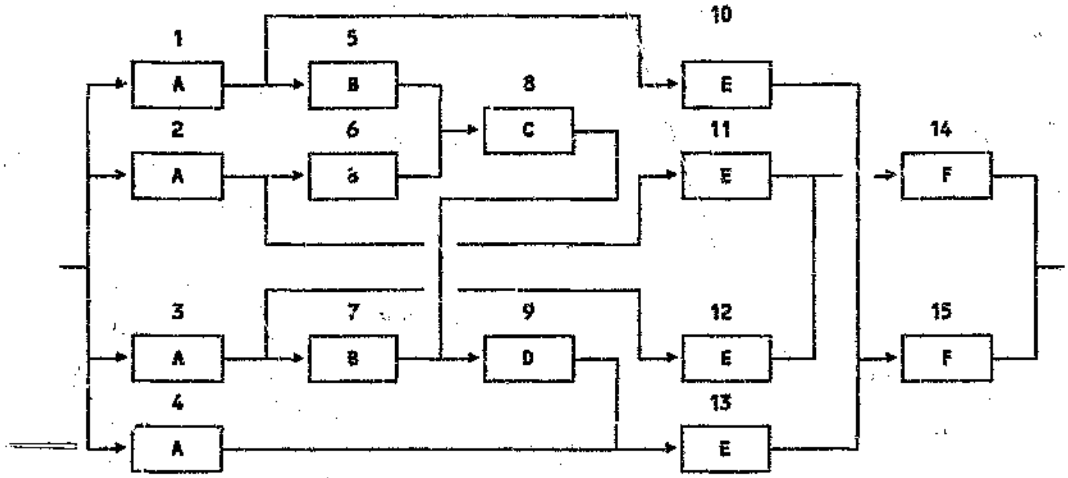

Figure 4.3.1 Boeing 747 115V AC Bus - RBD

Table 4.3.1 shows the physical items which make up a component and the overall MTBF of the component.

Table 4.3.1 Boeing 747 115V AC Bus Component Data

Comp	Description	MTBF (hrs)
A	Engine, Constant speed drive, Load controller, Generator differential protection control transformer, Generator, Generator control unit, Generator control current transformer	1 920
B	Generator circuit breaker, Synchronous bus differential protection control transformer, Bus tie breaker	200 000
C	Bus power control unit, 2 x synchronous bus differential protection control transformers, Split system breaker	111 111
D	Synchronous bus differential protection control transformer, Bus tie breaker, Bus power control unit	125 000
E	2 x generator differential protection control transformers, Circuit breaker, Relay	250 000
F	Switch unit	500 000

The system function **ISYSUP** is easily deduced from the tie sets contained in **Figure 4.3.1** and can be found in **Appendix D.** It is not presented here as it is quite lengthy. The analysis of the electrical power system is also subject to the following assumptions:

- All failures are independent of one another, e.g. a failure in one generator channel will not effect the other generator channels.

- Failures downstream of the buses do not effect the system.

- A bus is considered operable provided that at least one power source is available to it.

- The split system breaker is closed.

The system function **ISYSUP** was programmed into subroutine **SYST** and the component and general program control data were entered into **RAMIN**. The output **RAMOUT** and all the above files can be found in **Appendix D**. The results indicated a mean life of 3959.30 hours and a standard deviation of 2254.38 hours. The reliability at 1000 hours was estimated at 0.973. The empirical reliability distribution is shown in **Table 4.3.2**.

The file **TTFLIST** was used to generate a histogram of simulated system failure times (refer **Figure 4.3.2**). The histogram shows a skewed distribution.

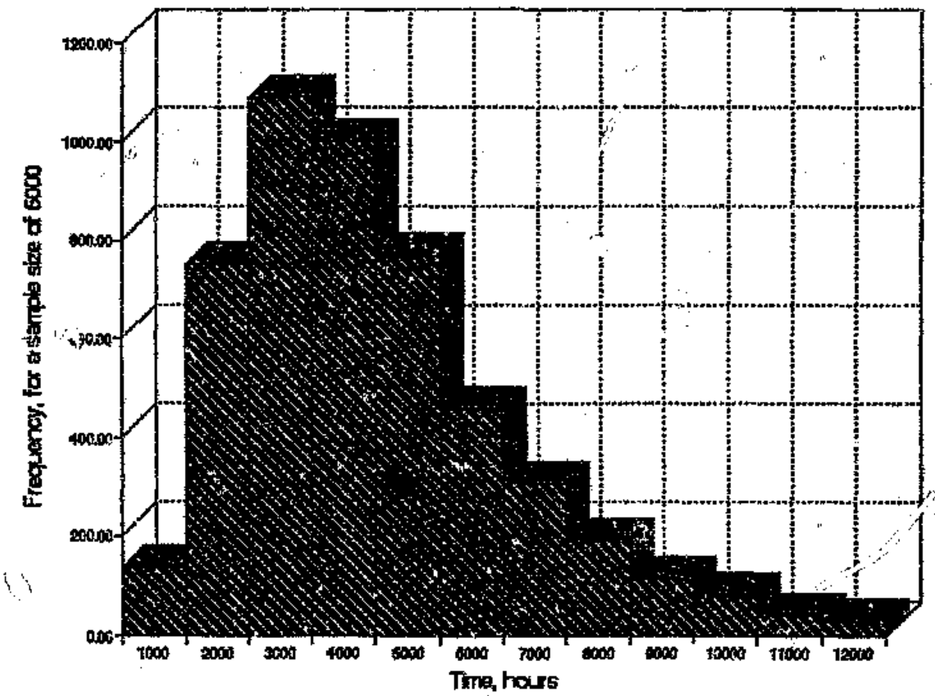

Figure 4.3.2 Histogram of System Failure Times - B747 115V AC Bus

The empirical point estimates of reliability were also computed using **AMIR®**. The subroutine **LBOUT** as well as the input and output files, i.e. **IRB1** and **OUTR1** can all be found in **Appendix E.** The subroutine **LBOUT** contained the system function. A sample size of 5000 was chosen together with 20 class intervals each of width 1000 hours.

The results indicated a mean life of 3934.62 hours which was close to that estimated by the simulator. Note, an empirical <u>un</u>reliability distribution was generated by **AMIR®** while the simulator generated a reliability distribution. For example, the unreliability at 1000 hours was estimated by **AMIR®** at 0.03, therefore the reliability at this time would be 0.97. The Percentage Relative Standard Deviation (PRSD) associated with this value was given as 8.097 percent. This can be interpreted in the sense that with probability 0.95, the exact unreliability lies in the interval:

$$(0.03-0.03\times2\times0.0809\ ,\ 0.03+0.03\times2\times0.0809) = (0.025\ ,\ 0.035) \qquad (4.3.1)$$

The reliability results obtained from the simulator were all subtracted from one to obtain unreliability, i.e. 1–RMLE. These results were then compared to the unreliability results obtained from **AMIR®** in **Table 4.3.2.** The simulator results compared favourably with the **AMIR®** values.

Table 4.3.2 Comparison of Simulator and AMIR* Results - B747 115V AC Bus

Time	Simulation Model results			AMIR* Results	
	R95L	RMLE	1-RMLE	F(t)	PRSD
1000	0.969	0.973	0.027	0.030	8.097
2000	0.814	0.823	0.177	0.185	2.964
3000	0.594	0.605	0.395	0.404	1.718
4000	0.394	0.405	0.595	0.604	1.145
5000	0.241	0.251	0.749	0.754	0.808
6000	0.150	0.159	0.841	0.845	0.605
7000	0.090	0.097	0.903	0.903	0.463
8000	0.054	0.059	0.941	0.941	0.353
9000	0.032	0.036	0.964	0.961	0.286
10000	0.017	0.020	0.980	0.976	0.224
11000	0.010	0.012	0.988	0.985	0.177
12000	0.005	0.006	0.994	0.990	0.139

4.4 SINGLE COMPONENT SYSTEM

This is a classic example which concerns the availability estimation of a single component. Dhillion[7] shows how a Markovian model can be developed to predict the steady state availability of a single component with constant failure and repair rates. He applied the model to various components in the power generation field such as condensers, generator units, etc.

This example allowed one to compare the availability estimate obtained by the simulator with the true value. It also allowed one to check the sampling functions of the simulator. The results of sampling from the exponential, normal and Weibull distributions were therefore compared with true values.

The system consists of only one component and the system function **ISYSUP** is simply:

$$ISYSUP = KST(1) \quad (4.4.1.1)$$

The estimation of availability and the checking of the sampling functions are described in the next two sections.

4.4.1 AVAILABILITY ESTIMATE

It will be assumed that the component availability is being measured as a two-state repairable system. In other words, the component can only be in an operational or failed state. This is an example of Inherent Availability where uptime consists of actual working time and downtime consists of unscheduled repair time (refer **Table 1.3.1**). It was also assumed, for the availability simulation, that all failure and repair rates are constant and that the repaired system is as good as new. The failure and repair rates were taken as 0.01 and 0.1 respectively, i.e. an exponential failure distribution with a MTBF of 100 hours and an exponential repair distribution with a MTTR of 10 hours.

Subroutine **SYST**, **RAMIN** and **RAMOUT** for the estimation of availability can all be found in **Appendix F**. As the system consists of only one component, it makes no difference whether the component is left on or switched off during system repair or whether the system repair time is the average, minimum or maximum of the component repair times. The steady state availability estimate obtained from 5000 histories was 0.91 and the standard deviation approached zero. The average uptime and downtime were 98.70 and 9.78 hours respectively.

Dhillon[7] used the Markov technique to calculate the steady state availability of a single generator. The formulae was presented in **Equation 2.62** as:

$$A_I(t) = \frac{\mu}{\mu+\lambda}$$

Substituting the failure and repair rates into the above formulae yields an availability of 0.91 which is the estimate obtained by the simulator.

4.4.2 CHECKING THE SAMPLING FUNCTIONS

The case of a single component allows one to check the results of sampling from various distribution functions. Samples from the exponential, normal and Weibull distributions were therefore compared with true values. Note, two cases were checked for the Weibull distribution, i.e. a decreasing and an increasing failure rate.

The variables in **RAMIN** were changed from a reliability/availability simulation to a pure reliability simulation. The probability distributions and associated parameters were also changed as required. Subroutine **SYST** remained the same as before. The output files for each distribution can be found in **Appendix F.**

The simulator results and the true values for each case were compared in **Tables 4.4.2.1** through **4.4.2.4.** The simulator results were close to the true values.

Table 4.4.2.1 Normal Distribution ($\mu = 100, \sigma = 30$)

Hours	R95L	RMLE	R True
10	0.998	0.999	0.998
20	0.996	0.997	0.996
30	0.992	0.994	0.990
40	0.973	0.977	0.977
50	0.946	0.951	0.951
60	0.893	0.900	0.908
70	0.821	0.830	0.841
80	0.728	0.739	0.745
90	0.601	0.612	0.629
100	0.473	0.484	0.500
110	0.344	0.355	0.371
120	0.230	0.240	0.255
130	0.141	0.150	0.159
140	0.078	0.084	0.092
150	0.038	0.043	0.049
160	0.017	0.020	0.023
170	0.007	0.009	0.010
180	0.002	0.003	0.004
190	0.000	0.001	0.002

Table 4.4.2.2 Exponential Distribution (MTBF = 100)

Hours	R95L	RMLE	R True
10	0.904	0.910	0.905
20	0.808	0.817	0.819
30	0.733	0.743	0.741
40	0.660	0.671	0.670
50	0.594	0.605	0.607
60	0.531	0.542	0.549
70	0.479	0.490	0.497
80	0.431	0.442	0.449
90	0.389	0.401	0.407
100	0.351	0.363	0.368
110	0.320	0.331	0.333
120	0.287	0.297	0.301
130	0.255	0.265	0.273
140	0.229	0.239	0.247
150	0.206	0.216	0.223
160	0.185	0.194	0.202
170	0.168	0.176	0.183
180	0.153	0.161	0.165
190	0.136	0.144	0.150
200	0.124	0.132	0.135
210	0.112	0.120	0.122
220	0.103	0.110	0.111
230	0.092	0.099	0.100
240	0.083	0.089	0.091
250	0.073	0.079	0.082

Table 4.4.2.3 Weibull Distribution (η = 100, β = 0.5)

Hours	R95L	RMLE	R True
10	0.719	0.729	0.729
20	0.626	0.637	0.639
30	0.561	0.573	0.578
40	0.516	0.528	0.531
50	0.474	0.486	0.493
60	0.442	0.454	0.461
70	0.415	0.426	0.433
80	0.390	0.402	0.409
90	0.371	0.382	0.387
100	0.351	0.363	0.368
110	0.333	0.344	0.350
120	0.322	0.333	0.334
130	0.309	0.320	0.320
140	0.293	0.304	0.306
150	0.278	0.288	0.294
160	0.264	0.274	0.282
170	0.253	0.264	0.271
180	0.243	0.253	0.261
190	0.235	0.245	0.252
200	0.224	0.234	0.243
210	0.217	0.227	0.235
220	0.211	0.221	0.227
230	0.203	0.213	0.219
240	0.196	0.205	0.212
250	0.189	0.199	0.206

Table 4.4.2.4 Weibull Distribution (η = 100, β = 2)

Hours	R95L	RMLE	R True
10	0.989	0.991	0.990
20	0.959	0.963	0.961
30	0.913	0.919	0.914
40	0.846	0.854	0.852
50	0.768	0.778	0.779
60	0.687	0.698	0.698
70	0.599	0.610	0.613
80	0.513	0.524	0.527
90	0.427	0.438	0.445
100	0.351	0.363	0.368
110	0.283	0.294	0.298
120	0.219	0.228	0.237
130	0.170	0.179	0.185
140	0.128	0.136	0.141
150	0.098	0.105	0.105
160	0.067	0.073	0.077
170	0.047	0.052	0.056
180	0.033	0.037	0.039
190	0.022	0.026	0.027
200	0.015	0.018	0.018
210	0.009	0.012	0.012
220	0.006	0.008	0.008
230	0.004	0.005	0.005
240	0.002	0.003	0.003
250	0.001	0.002	0.002

4.5 PRODUCTION LINE SYSTEM

This example concerns the estimation of the steady state availability of a typical production line. The example was extracted from a set of examples compiled by Dubi[8] and modified to include non-exponential failure and repair distributions.

The example demonstrates various functions of the simulation model, i.e. estimating the availability of systems with non-exponential failure and repair densities, leaving components on or switching them off during system repair, as well as different ways of calculating system repair time. The following types of simulation runs were completed for the production line:

(1) Components were switched off during system repair and the system repair time was equal to the sum of component repair times. For this case, a number of different sample sizes were also selected for illustrative purposes.

(2) Components were switched off during system repair and the system repair time was equal to the maximum component repair time.

(3) Components were left on during system repair and the system repair time was equal to the sum of component repair times.

(4) Components were left on during system repair and the system repair time was equal to the maximum component repair time. The simulator results for this run were validated using **SPAR**®.

(5) Pure reliability simulation. This was done in order to compare the estimates of maintained and non-maintained system reliability.

The production line contains redundancies and will continue to operate when some of its components have failed. The line contains three processes in active redundancy. In other words, only one process needs to be operating for the line to be operational. The availability of the line is further enhanced by 3 components in parallel, any of which can be used by any one process. The reliability network of the production line is shown in **Figure 4.5.1.**

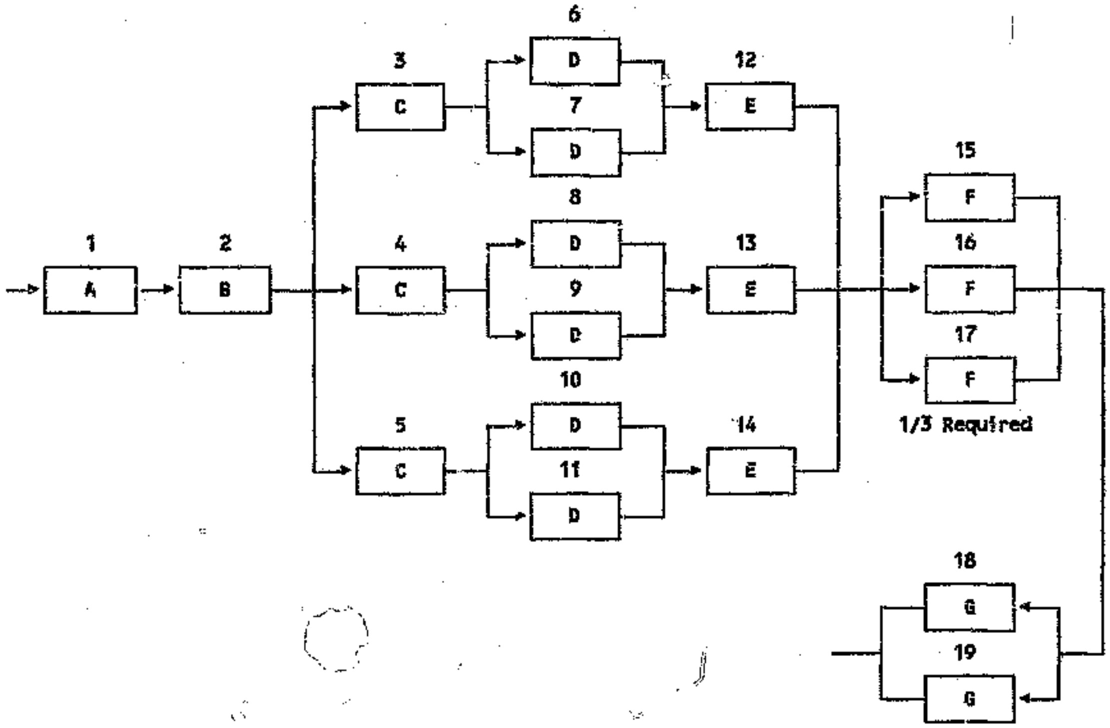

Figure 4.5.1 Production Line System - RBD

Nowlan and Heap[18] explain that it is usually only items with one failure mode or a dominant failure mode that benefit from preventive maintenance. Most complex components (those with many failure modes) exhibit non-age related failure times, i.e. they have negative exponential failure densities and should not be subject to preventive maintenance (refer **Section 2.2**).

Most items on the production line are complex without a dominant failure mode and are therefore not subject to preventive maintenance. The maintenance policy for the production line is therefore to repair the system when it breaks, i.e. components are only repaired following a system failure.

The production line operates continuously, i.e. 24 hours per day and 7 days per week. The line therefore does not have a standby state because when it is operational it is always working (refer **Table 1.3.1**). Production line downtime therefore consists only of corrective maintenance time as no preventive maintenance is carried out. It will also be assumed that there is no delay in repair due to a shortage of manpower, spares, etc. The above situation exactly describes that of Inherent Availability which was discussed in **Section 1.3.1**.

Production line components exhibited negative exponential and Weibull failure densities and normal repair time densities. The properties of each component are shown in **Table 4.5.1**.

Table 4.5.1 Production Line Component Data

Comp	Time to Failure (Days)			Time to Repair (Days)		
	Distr	Param 1	Param 2	Distr	Param 1	Param 2
A	Weib	1140.63	0.9	Nrml	2.92	0.50
B	Expn	1520.96	N/A	Nrml	2.92	0.50
C	Expn	152.21	N/A	Nrml	12.05	2.00
D	Expn	45.55	N/A	Nrml	24.09	4.00
E	Weib	101.47	1.1	Nrml	12.05	2.00
F	Expn	202.58	N/A	Nrml	12.05	2.00
G	Expn	182.50	N/A	Nrml	4.02	0.67

The system function **ISYSUP** is very large, i.e. 36 tie sets in all, and is therefore not displayed here. The reader is referred to **Appendix G** for the system function. Note, it is possible to reduce the number of tie sets to 3 by defining groups of components as cat components. This was not done because it was not possible to define cat components like this in **SPAR**[*].

Subroutine **SYST** and **RAMIN** for run number 1 can be found in **Appendix G.** A sample size of 5000 was selected for all simulations runs and additional sample sizes of 10, 50 and 100 were also selected for run number 1. The empirical reliability distribution was defined as having 15 class intervals each of width 10 days. **RAMOUT** for each simulation can also be found in **Appendix G.**

Sample sizes for run number 1 of 10, 50 and 100 show that the availability value converges rapidly to that of steady state. The average availability is 0.22 for a sample size of 10, 0.27 for a sample size of 50 and 0.27 for a sample size of 100. The standard deviation is 0.04 for 10 histories which reduces to 0.02 at 50 histories and 0.01 at 100 histories. The 95 % confidence levels for availability at the above histories are 0.18, 0.25 and 0.25. At a sample size of 5000 the steady state availability value is 0.28 as is the 95 % confidence limit. For this sample size, the standard deviation is very small.

For run numbers 2, 3 and 4 (sample sizes of 5000 in all cases) the steady state availabilities were found to be 0.66, 0.10 and 0.60. The values differed significantly due to the specification of system repair time and whether components were left on or switched off during system repair.

For each run number (excluding run number 5) the lists of system repair times **TTRLIST** were used to generate histograms. A comparison of these histograms are shown in **Figure 4.5.3**. Note, sample sizes of 5000 were used in all cases.

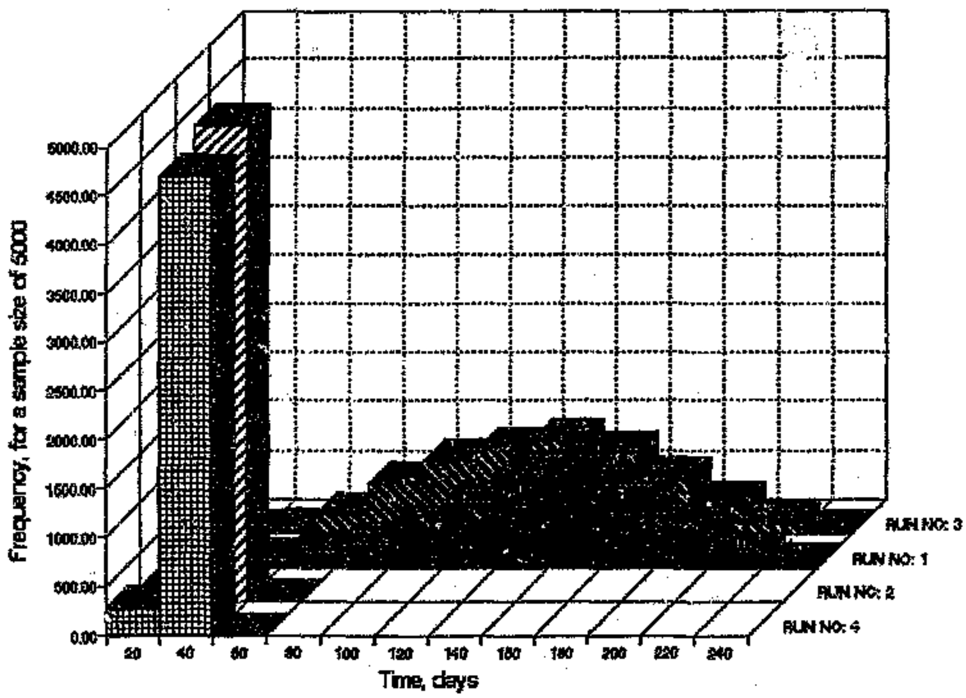

Figure 4.5.3 Histograms of System Repair Times – Production Line System

For run numbers 1 and 3 the mean repair time was 135.40 and 123.92 respectively with standard deviations of 39.97 and 38.43. Whereas, for runs 2 and 4 the mean repair time was 27.68 and 27.05 respectively with standard deviations of 4.01 and 4.71.

Run numbers 1 and 3 show a much wider dispersion than 2 and 4. The system repair times for runs 2 and 4 are dominated by a particular repair mode as the system repair time is equal to the maximum component repair time. On the other hand, the system repair times for runs 2 and 4 are more spread out as the system repair time is equal to the sum of component repair times.

Note, selecting the system repair time equal to the sum of component repair times means having components repaired one after the other following a system failure. Selecting the system repair time equal to the maximum component repair time means having all components repaired simultaneously following a system failure.

For run numbers 1, 3, 4, and 5 the lists of system failure times **TTFLIST** were used to generate histograms (refer **Figure 4.5.4**). Note, the histograms for run numbers 1 and 2 are the same. Sample sizes of 5000 were used in all cases.

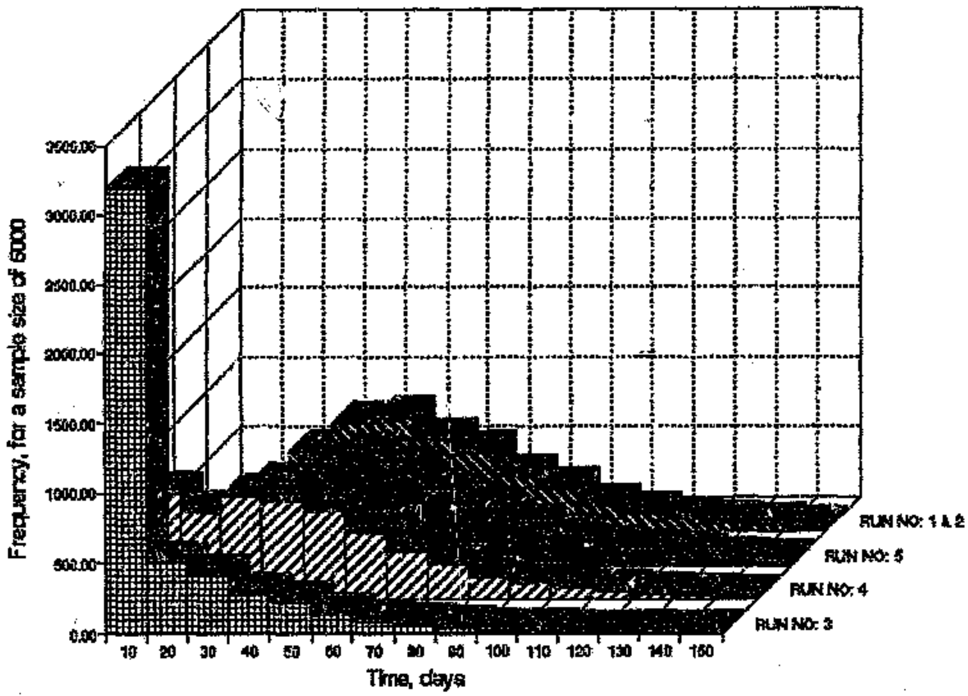

Figure 4.5.4 Histograms of System Failure Times - Production Line System

For run numbers 1, 3, 4 and 5 the mean failure time was 53.67, 13.86, 40.12 and 54.38 respectively with standard deviations of 28.87, 22.52, 30.07 and 29.74. Note, the values for run 2 are the same as those of run 1.

There are differences when comparing the histograms of run numbers 1, 3 and 4 with run number 5. These differences are to be expected as the histograms of runs 1, 3 and 4 are based on the system times to fail from the last repair, whereas run number 5 is not. Differences also occur between runs 1, 3 and 4 themselves which can be attributed to the repair specification and whether components are switched off or left on during system repair. Note, although the repair specification of runs 1 and 2 are different, the components are switched off during system repair, hence the histograms are the same.

SPAR® was used to validate the availability estimate for run number 4. In order for the two models to be comparable certain options had to be selected in **SPAR®**, i.e.

- The system checkup level was selected. If the checked system is found operational, then no further checking is done. Only when the system is found failed will components be checked and repaired.

- The continuous mode was selected for the system checkup level. In this mode the system is checked at each stochastic event.

- All components were defined as being repairable at level A. This means that we are not taking into account the effects of spare parts and turn around times for off equipment repair.

- The default settings, i.e. components remain active during system repair and the system repair time is equal to the maximum component repair time were left unchanged.

- All components with exponential distributions were specified as having Weibull distributions with $\beta=1$. This was done to ensure that component times to failure were based on the system repair times.

- The same system function that was entered into the simulator was entered into **SPAR®**. It would have been possible to simplify the system function entered into the simulator by defining groups of components as cat components. The reason why this was not done is that the repair time would then be defined for the cat component as a whole, whereas in **SPAR®** it would be defined per subcomponent.

The **SPAR®** input and output files, i.e. **STIN** and **STOUT**, as well as the subroutine **LBOUT** can all be found in **Appendix H**. This subroutine contains the system function. A sample size of 500 was chosen and each sample consisted of 10000 days of utilisation. An average availability of 0.61 and a Percent Relative Standard Deviation of 3.49 % (PRSD) was obtained by **SPAR®**. The PRSD is the statistical error in the sense that with probability 0.95, the exact answer lies in the interval:

$$(0.6151-0.6151\times2\times0.0349\ ,\ 0.6151+0.6151\times2\times0.0349) = (0.5722\ ,\ 0.6580) \quad (4.5.1)$$

The average availability of 0.60 obtained by the simulator compares favourably

with the above values obtained by SPAR[®].

5 CONCLUSIONS

Engineers require an economical, accurate and easy to use system reliability and availability tool. The simulator developed in this report is one such tool.

Analytical methods for the reliability and availability estimation of large systems often become impractical. The Markov method for system availability estimation leads to large numbers of simultaneous differential equations. The situation is further complicated with the inclusion of non-exponential failure and repair densities. Fortunately, the Monte Carlo simulation approach allows one to easily solve such problems.

The simulator is based to a large extent on the work conducted by Forry[13] with the inclusion of the system function concept pioneered by Dubi[9]. The simulator is limited to the analysis of those systems whose logic can be fully represented by a Reliability Block Diagram.

The Monte Carlo methodology was applied to simulate the failure and repair times of system components from their applicable probability distributions. From this, one could compute the time to failure and repair of systems which are made up of components. The systems analyzed included active redundant systems, standby systems, parallel and serial systems, as well as combinations of all of these. The simulator also gave one the option of leaving components on or switching them off during system repair. Different options were also available for calculating the system repair time.

The simulator generates an empirical reliability distribution as well as an estimate of average availability if required. Confidence levels and measures of dispersion are attached to each value. Raw data files containing system repair and failure times are made available for exporting to statistical software such as **STATGRAPHICS***. The raw data can be manipulated in these packages and a distribution can be fitted.

The program was written in **Microsoft FORTRAN 4.1*** but should work with most FORTRAN compilers. This allows one to run the simulator on a large variety of computers. The random number generator was purposely chosen to work with a very small computer word size which also allows one to run the simulator on a large variety of computers. An engineer with a basic knowledge of FORTRAN could easily use the simulator.

The simulator can accommodate 25 components or cat components. The cat components can contain a maximum of 5 subcomponents each. A maximum of 5000 histories are permitted as well as a maximum of 25 entries for the empirical reliability distribution table. These limits can be easily changed by modifying the appropriate dimension statements.

Based on the results of the simulation examples, it can be concluded that the simulator correctly determines the empirical reliability distribution as well as the average availability. All the validations completed in this study checked out.

The model does have limited applicability in terms of modelling general systems. To include general systems would require extensive modifications to the simulator. For the modelling of general systems it is best to resort to **AMIR®** or **SPAR®**.

It can be concluded that the simulator offers an economical, practical and accurate manner of estimating the reliability and availability of complex network systems. It can save the engineer many hours of tedious work by providing quick estimates of system reliability and availability.

APPENDIX A

PROGRAM SOURCE CODE LISTING

PROGRAM MAIN

```
      PROGRAM MAIN
C
C  ****************************************************
C  * GENERAL RELIABILITY/AVAILABILITY SYSTEM SIMULATOR *
C  ****************************************************
C
      COMMON /CRES/ RES,RESQ,NTYPE
      COMMON RMAXTM,RMINTM
C
C  ***************************************************************
C  * COMMON BLOCKS DEFINED FOR SUBROUTINE TAB                    *
C  * RES      -  SUM OF SYSTEM REPAIR TIMES                      *
C  * RESQ     -  SUM OF THE SQUARES OF SYSTEM REPAIR TIMES       *
C  * NTYPE=0 - ONLY RELIABILITY CALCULATION                      *
C  * NTYPE=1 - AVAILABILITY CALC, COMPONENTS SWITCHED OFF        *
C  *             DURING REPAIR                                   *
C  * NTYPE=2 - AVAILABILITY CALC, COMPONENTS LEFT ON             *
C  *             DURING REPAIR                                   *
C  ***************************************************************
C
      DIMENSION T(20),KFDN(20),FPTR(20,2),ICODE(20),PT(10)
      DIMENSION TSYSF(5000),RPTR(20,2),REP(3)
      DIMENSION KFREQ(25),KST(20),NF(20),TSYSR(5000)
C
C ***************************************************************
C * MAXIMUM NUMBER OF COMPONENTS IS 20                          *
C * MAXIMUM NUMBER OF HISTORIES IS 5000                         *
C * MAXIMUM NUMBER OF CLASS INTERVALS FOR EMPIRICAL RELIABILITY*
C * TABLE IS 25                                                 *
C * MAXIMUM NUMBER OF SUBCOMPONENTS FOR CAT COMPONENT IS 5      *
C *                                                             *
C * T     - TIME TO FAIL FOR COMPONENT OR CAT COMPONENT         *
C *         RETURNED BY FAILT AND/OR MODIFIED IN PROGRAM MAIN   *
C *         IF COMPONENT IS LEFT ON DURING SYSTEM REPAIR ETC    *
C * KFDN  - FAILURE DISTRIBUTION IDENTITY (ONE NUMBER)          *
C * FPTR  - FAILURE DISTRIBUTION PROPERTY (TWO NUMBERS)         *
C * ICODE - SINGLE COMP (0),ACTIVE STANDBY eg 2 out 5 (52)      *
C *         NON-ACTIVE STANDBY eg (-2)                          *
C * PT    - ARRAY PT(K) USED IN SUBROUTINE FAILT                *
C *         FOR CAT COMPONENTS                                  *
C *         INCLUDED IN PROGRAM MAIN FOR DIMENSIONING PURPOSES *
C * TSYSF - SYSTEM FALURE TIME                                  *
C * TSYSR - SYSTEM REPAIR TIME                                  *
C * KRDN  - REPAIR DISTRIBUTION IDENTITY (ONE NUMBER)           *
C * RPTR  - REPAIR DISTRIBUTION PROPERTY (TWO NUMBERS)          *
C * REP   - REP(KFIX) SYSTEM REPAIR TIME                        *
C * KFREQ - THE NUMBER OF SYSTEM FAILURES WITHIN EACH CLASS     *
C *         INTERVAL (EMPIRICAL RELIABILITY TABLE)              *
C * KST   - COMPONENT STATUS USED IN SUBROUTINE SYST            *
C * TEMP2 - VECTOR OF POTENTIAL SYSTEM FAILURE TIMES USED IN    *
C *         SUBROUTINE SYST                                     *
C * NF    - NUMBER OF FAILURES PER COMPONENT,CALCULATED IN      *
C *         SUBROUTINE SYST                                     *
C *                                                             *
C * RAMIN   - INPUT DATA FILE (UNIT 9)                          *
C * RAMOUT  - OUTPUT DATA FILE (UNIT 3)                         *
C * TTFLIST - OUTPUT DATA FILE (UNIT 4)                         *
C * TTRLIST - OUTPUT DATA FILE (UNIT 8)                         *
C * AVALIST - OUTPUT DATA FILE (UNIT 7) IF REQUIRED REMOVE      *
C *           COMMENT CHARACTERS                                *
```

```
C ****************************************************************
C
      OPEN (UNIT=9,FILE='RAMIN',STATUS='OLD',ACCESS='SEQUENTIAL'
     *,FORM='FORMATTED')
C
      READ (UNIT=9,FMT=117) NTYPE,NTIME,N,KFIX,IPROB,ISZE,FI
117   FORMAT (8X,I1,8X,I4,4X,I2,7X,I2,8X,I2,7X,I2,5X,F10.2)
C
C  ***********************************************************
C  * NTIME  - SAMPLE SIZE (NUMBER OF HISTORIES)               *
C  * N      - NO. OF COMPONENTS                               *
C  * ISZE   - NUMBER OF CLASS INTERVALS (EMPIRICAL            *
C  *          RELIABILITY DISTRIBUTION)                       *
C  *          AS INPUT BY THE USER                            *
C  * KFIX   - REPAIR SPEC                                     *
C  * KFIX=1 - SYSTEM REPAIR TIME EQUAL TO SUM OF COMPONENT    *
C  *          REPAIR TIMES                                    *
C  * KFIX=2 - SYSTEM REPAIR TIME EQUAL TO LARGEST COMPONENT   *
C  *          REPAIR TIME                                     *
C  * KFIX=3 - SYSTEM REPAIR TIME EQUAL TO AVERAGE COMPONENT   *
C  *          REPAIR TIME                                     *
C  * FI     - CLASS INTERVAL WIDTH AS INPUT BY USER           *
C  * IPROB  - RUN IDENTIFICATION NUMBER                       *
C  ***********************************************************
C
      DO 20 I=1,N
       READ (UNIT=9,FMT=205) ICODE(I),KFDN(I),FPTR(I,1),FPTR(I,2)
     *,KRDN(I),RPTR(I,1),RPTR(I,2)
20    CONTINUE
205   FORMAT (8X,I2,/,7X,I2,10X,E14.7,10X,E14.7,/,
     *7X,I2,10X,E14.7,10X,E14.7)
C
C ****************************************************************
C
      OPEN (UNIT=3,FILE='RAMOUT',STATUS='OLD',ACCESS='SEQUENTIAL'
     *,FORM='FORMATTED' )
C
      WRITE (UNIT=3,FMT=118) IPROB
118   FORMAT (/,26X,'RUN NO. ',I2,//,26X,'INPUT BLOCK',/)
C
      WRITE (UNIT=3,FMT=120) NTYPE,NTIME,N,KFIX,ISZE,FI
120   FORMAT (2X,'TYPE OF RUN (0,1,OR 2)...........................'
     *,I7,/,
     *2X,'REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......',I7,/,
     *2X,'NUMBER OF COMPONENTS IN THE SYSTEM.................',I7,/,
     *2X,'REPAIR SPECIFICATION (1,2 OR 3)....................',I7,/,
     *2X,'NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....',I7,/,
     *2X,'CLASS INTERVAL WIDTH...............................',F10.2,/)
C
      DO 123 I=1,N
      WRITE (UNIT=3,FMT=119) I,ICODE(I),KFDN(I),FPTR(I,1)
     *,FPTR(I,2),KRDN(I),RPTR(I,1),RPTR(I,2)
123   CONTINUE
119   FORMAT (2X,'COMP NO.',I2,1X,'ICODE..',I2,/,13X,'KFDN..'I2
     *,2X,'FPTR(1)..',E14.7,2X,'FPTR(2)..',E14.7,/,13X,'KRDN..',I2
     *,2X,'RPTR(1)..',E14.7,2X,'RPTR(2)..',E14.7,2X/)
C
      WRITE (UNIT=3,FMT=122)
122   FORMAT (/,27X,'OUTPUT BLOCK',/)
C
C ****************************************************************
C
      OPEN (UNIT=4,FILE='TTFLIST',STATUS='OLD',ACCESS='SEQUENTIAL'
     *,FORM='FORMATTED')
C
      OPEN (UNIT=8,FILE='TTRLIST',STATUS='OLD',ACCESS='SEQUENTIAL'
     *,FORM='FORMATTED')
C
C      OPEN (UNIT=7,FILE='AVALIST',STATUS='OLD',ACCESS='SEQUENTIAL'
C     *,FORM='FORMATTED')
C
```

```
      WRITE (6,127)
127   FORMAT (/////,5X,'RAM SIMULATION MODEL FOR NETWORK SYSTEMS',/,
     *5X,'          PREPARED BY RAY JENKINS',////////////)
C
C ********************************
C * INITIALIZATION OF PARAMETERS *
C ********************************
C
      MS=1
      RES=0
      MSW=1
      CLOCK=0
      UPT=0
      RMAXTM=0
      RMINTM=.1E+9
      DO 64 I=1,N
         NF(I)=0
64    CONTINUE
C
C  ***************************************************************
C  * MS     - PROGRAM CONTROL VARIABLE USED IN SUBROUTINE TAB, SET *
C  *          TO 1 IN MAIN AND SET AGAIN TO ZERO IN SUBROUTINE TAB *
C  * MSW    - PROGRAM CONTROL VARIABLE, BEST EXPLAINED FURTHER ON  *
C  *          WHERE IT IS USED                                     *
C  * CLOCK  - SUMMATION OF SYSTEM TIMES TO FAILURE AND REPAIR      *
C  * UPT    - SUMMATION OF SYSTEM UPTIMES                          *
C  * RMAXTM - MAXIMUM OF SYSTEM FAILURE TIMES                      *
C  * RMINTM - MINIMUM OF SYSTEM FAILURE TIMES                      *
C  * NF     - NUMBER OF FAILURES PER COMPONENT                     *
C  ***************************************************************
C
C  *********************************************
C  * OBTAIN SAMPLE (NTIMES) FOR SYSTEM FAILURE *
C  *********************************************
C
      DO 50 I=1,NTIME
      WRITE (6,*) I
C
C ************************************************************
C * MSW - PROGRAM CONTROL VARIABLE                           *
C *       IF AVAILABILITY CALCULATION IS SPECIFIED, THEN THE *
C *       PROGRAM SHALL CALL FAILT ONCE IMMEDIATELY BELOW    *
C *       WHILST ALL OTHER CALLS SHALL TAKE PLACE            *
C *       WHERE SUBROUTINE FAILT IS SPECIFIED A SECOND TIME  *
C ************************************************************
C
C ************************************************
C * OBTAIN RANDOM TIMES TO FAIL FOR EACH COMPONENT *
C ************************************************
C
      CALL FAILT (T(J),KFDN(J),FPTR(J,1),FPTR(J,2),ICODE(J),PT)
C
C ************************************************************
C *       SUBROUTINE FAILT RETURNS TIME TO FAIL T            *
C *                                                          *
C * T  - TIME TO FAIL FOR ORDINARY COMPONENT OR              *
C *      CAT COMPONENT (ACTIVE OR NON ACTIVE STANDBY)        *
C *      ALWAYS GREATER THAN ZERO                            *
C * PT - ARRAY USED IN SUBROUTINE FAILT, STBY AND PARL TO    *
C *      CALCULATE T FOR CAT COMPONENTS                      *
C ************************************************************
C
15    CONTINUE
C
C *********************************
C * COMPUTE SYSTEM TIME TO FAILURE *
C *********************************
C
60    CALL SYST (SYSF,T,N,KST,TEMP2,NF)
C
C ************************************************************
```

```
C * FUNCTION RETURNS SYSTEM FAILURE TIME (SYSF)                  *
C *                                                             *
C * T,N   - ARE SUPPLIED BY PROGRAM MAIN                        *
C * T     - COMPONENT TIME TO FAIL                              *
C * N     - NUMBER OF COMPONENTS                                *
C * KST   - COMPONENT STATUS, USED IN FUNCTION SYS              *
C * TEMP2 - VECTOR OF POTENTIAL SYSTEM FAILURE TIMES USED       *
C *         IN SUBROUTINE SYST                                  *
C * NF    - TOTAL NUMBER OF FAILURES PER COMPONENT              *
C **************************************************************
C
      IF (SYSF.GT.RMAXTM) RMAXTM=SYSF
      IF (SYSF.LT.RMINTM) RMINTM=SYSF
C
C ****************************************************************
C * FOR NTIME HISTORIES THE MAXIMUM SYSTEM FAILURE TIME SHALL *
C * BE STORED IN RMAXTM AND THE MINIMUM IN RMINTM               *
C ****************************************************************
C
      TSYSF(I)=SYSF
      IF (SYSF.EQ.0.) TSYSF(I)=.1E-7
      IF (NTYPE.EQ.0) GOTO 45
C
C *********************************************************************
C *********************************************************************
C *********************************************************************
C *********************************************************************
C * AVAILABILITY CALCULATION                                         *
C *                                                                   *
C * FOR THR RELIABILITY CALCULATION, GOTO 45                          *
C *                                                                   *
C * REPAIR FAILED SYSTEM AND ESTIMATE AVAILIBILITY                    *
C *                                                                   *
C * UPON CALL BY USER IN NTYPE SPEC, AND UPON SYSTEM FAILURE          *
C * THIS PROGRAM DETERMINES A RANDOM TIME TO REPAIR EQUAL TO THE      *
C * SUM OF THE TIMES TO REPAIR, THE MAX OF REPAIR TIMES OR THE MEAN   *
C * REPAIR TIME OF COMPONENTS FAILED AT OR BEFORE THE SYSTEM FAILURE *
C * TIME, DEPENDING ON USERS SPEC IN KFIX, USER MAY ALSO SPECIFY      *
C * WHETHER THE CLOCK SHOULD RUN OR STOP DURING REPAIR. SYSTEM        *
C * AVAILABILITY IS ESTIMATED AND PRINTED OUT WITH CLOCK TIME,        *
C * SYSTEM FAIL TIME AND REPAIR TIME                                  *
C *                                                                   *
C * COMPONENT FAILURES ARE REPAIRED WHEN A SYSTEM FAILURE OCCURS      *
C *                                                                   *
C * LOGIC:                                                            *
C * 1. IF A COMPONENT HAS FAILED BEFORE OR AT SYSTEM FAILURE TIME     *
C *   1.1 GENERATE A NEW TIME TO FAILURE FOR THE COMPONENT            *
C *   1.2 GENERATE A TIME TO REPAIR FOR THE COMPONENT                 *
C *   1.3 FIND THE SYSTEM REPAIR TIME BASED ON REPAIR TIME OF         *
C *       COMPONENTS                                                  *
C *                                                                   *
C * 2. IF A COMPONENT HAS NOT FAILED BY SYSTEM FAILURE TIME           *
C *   2.1 COMPONENTS LEFT SWITCHED ON DURING SYSTEM REPAIR            *
C *       2.1.1 COMPONENT HAS STILL NOT FAILED BY THE END OF SYSTEM   *
C *             REPAIR TIME, CARRY THE REMAINING TIME LEFT ON THE     *
C *             COMPONENT TO NEXT ITERATION                           *
C *       2.1.2 COMPONENT FAILED WHILE THE SYSTEM WAS BEING REPAIRED *
C *             OR WHEN IT WAS REPAIRED, COMPONENT TIME TO FAILURE    *
C *             SET TO ZERO FOR NEXT ITERATION                        *
C *             WHEN WE TRY TO SWITCH THE SYSTEM ON IT MAY            *
C *             IMMEDIATELY FAIL DEPENDING ON ISYSUP                  *
C *   2.2 COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR                *
C *       2.2.1 THE REMAINING TIME LEFT ON THE COMPONENT IS CARRIED   *
C *             FORWARD TO NEXT ITERATION                             *
C *                                                                   *
C * NOTE:                                                             *
C * IF THE COMPONENTS ARE SWITCHED OFF OR LEFT ON DURING SYSTEM       *
C * REPAIR, THE AGE OF THE COMPONENT IS NOT LOST.                     *
C * IT IS THEREFORE QUITE LOGICAL TO USE THE NON EXPONENTIAL          *
C * FAILURE DISTRIBUTIONS SUCH AS WEIBULL AND NORMAL                  *
C *********************************************************************
```

```
C
      REP(1)=0
      REP(2)=0
      REP(3)=0
      NREP=0
C
C ***********************************************************************
C * REP(KFIX)    - SYSTEM REPAIR TIME WHICH DEPENDS ON KFIX              *
C * REPAIR SPEC - KFIX                                                   *
C * NREP        - NUMBER OF COMPONENTS REPAIRED AT EACH SYSTEM FAILURE *
C *               USED TO CALCULATE AVERAGE REPAIR TIME                  *
C ***********************************************************************
C
      MI=NTYPE-1
C
C **********************************************************************
C * NTYPE = 1 AVAIL CALC COMPONENTS SWITCHED OFF DURING REPAIR (MI=0) *
C *       = 2 AVAIL CALC COMPONENTS SWITCHED ON DURING REPAIR  (MI=1) *
C *       = 0 ONLY FOR RELIABILITY NOT APPLICABLE HERE                  *
C * MI IS A PROGRAM CONTROL VARIABLE, RELATES TO WHETHER                *
C * COMPONENTS ARE SWITCHED ON OR OFF DURING REPAIR AND                 *
C * CAN BE EITHER 0 OR 1                                                *
C * UPT   - SYSTEM UPTIME                                               *
C * CLOCK - SYSTEM UPTIME + DOWNTIME                                    *
C **********************************************************************
C
      CLOCK=CLOCK+SYSF
      UPT=UPT+SYSF
C
      DO 317 K=1,N
            T(K)=T(K)-SYSF
C
C *******************************************************************
C * THIS LOOP IS ACCOMPLISHED FOR EACH COMPONENT FOR EACH HISTORY *
C * SUBTRACT SYSTEM FAILURE TIME FROM COMPONENT FAILURE            *
C * TIMES                                                          *
C *******************************************************************
C
            IF (T(K)) 305,305,303
C
C *********************************************
C * FOR EACH COMPONENT                        *
C * T(K) < 0 , =0 GOTO 305 (COMPONENT FAILED) *
C * T(K) > 0 , GOTO 303 (COMPONENT NOT FAILED) *
C *********************************************
C
303         IF (MI) 315,315,304
C
C **************************************************************
C * PROGRAM PASSES THROUGH HERE IF COMPONENTS ARE LEFT ON     *
C * DURING REPAIR, NTYPE=2, MI=1, GOTO 304                     *
C * IF COMPONENTS SWITCHED OFF DURING REPAIR GOTO 315          *
C **************************************************************
C
            GOTO 317
C
            CALL FAILT (T(K),KFDN(K),FPTR(K,1),FPTR(K,2),
     *      ICODE(K),PT)
C
            REP(1)=REP(1)+RDUM
            NREP=NREP+1
            REP(2)=AMAX1(REP(2),RDUM)
317         CONTINUE
C
      REP(3)=REP(1)/NREP
      RES=RES+REP(KFIX)
      RESQ=RESQ+REP(KFIX)*REP(KFIX)
C
C ****************************************************
C * T - TIME TO FAIL,ALWAYS GREATER THAN ZERO        *
C * RDUM - TIME TO REPAIR ALWAYS GREATER THAN ZERO   *
```

```
C * RES & RESQ - PARAMATERS IN THE COMMON BLOCK /CRES/ *
C * USED IN SUBROUTINE TAB                             *
C *****************************************************
C
      CLOCK=CLOCK+REP(KFIX)
      AVAL=UPT/CLOCK
C
C      WRITE (UNIT=7,FMT=348) CLOCK,SYSF,REP(KFIX),AVAL
C348   FORMAT (2X,F10.2,5X,F10.2,5X,F10.2,5X,F5.3)
C
C ******************************************************************
C * AVAL - CUMULATIVE AVAILABILITY THROUGHOUT THE SIMULATION       *
C *                                                                *
C * THE FINAL VALUE OF AVAL IS ALSO CALCULATED IN SUBROUTINE TAB   *
C * USING AVERAGE SYSTEM FAILURE AND REPAIR TIMES                  *
C * THIS IS THE AVERAGE AVAILABILITY VALUE IN RAMOUT               *
C ******************************************************************
C
311   IF (MI) 320,320,307
C
C ****************************************************************
C * PROGRAM DOES THIS LOOP IF COMPONENTS ARE LEFT ON DURING      *
C * REPAIR, NTYPE=2, MI=1                                        *
C ****************************************************************
C
307   DO 312 K=1,N
C
C ********************************************************************
C * T(K) < 0                                                         *
C * PROGRAM PASSES THROUGH HERE IF COMPONENT FAILURE TIME IS LARGER *
C * THEN SYSTEM FAILURE TIME                                         *
C * REP(KFIX) - SYSTEM REPAIR TIME                                   *
C * FAILURES ARE REPAIRED WHEN A SYSTEM FAILURE OCCURS               *
C *                                                                  *
C * IF THE TIME LEFT ON A COMPONENT:                                 *
C *    IS LARGER THAN THE SYSTEM REPAIR TIME - CARRY FORWARD         *
C *    REMAINING TIME                                                *
C *    IS EQUAL TO SYSTEM REPAIR TIME - REMAINING TIME IS ZERO       *
C *    ANYWAY                                                        *
C *    IS LESS THAN SYSTEM REPAIR TIME - REMAINING TIME IS SET TO    *
C *    ZERO                                                          *
C ********************************************************************
C
308          T(K)=T(K)+REP(KFIX)
             IF (T(K)) 309,312,310
             GOTO 312
310          T(K)=0
312   CONTINUE
C
45    CONTINUE
C
50    CONTINUE
C
C ***************************************************************
C ***************************************************************
C ***************************************************************
C *                      TABULATE RESULTS                       *
C ***************************************************************
C
      DO 51 I=1,NTIME
         CALL TAB (TSYSF(I),FI,NTIME,KFREQ,ISZE,MS)
C
C ************************************************************
C * EMPIRICAL STATS CALCULATED AND PRINTED BY THIS           *
C * SUBROUTINE                                               *
C * NTIME - NUMBER OF HISTORIES                              *
C ************************************************************
C
C
C ******************************************
C * NF - NUMBER OF FAILURES PER COMPONENT *
```

```
C *       CALCULATED BY SUBROUTINE SYST      *
C ******************************************
C
      WRITE (UNIT=3,FMT=126)
126   FORMAT (//,2X,13HFAILURE BLOCK,//,
     *2X,13HCOMPONENT NO.,5X,18HNUMBER OF FAILURES,/)
      DO 54 I=1,N
        WRITE (UNIT=3,FMT=125) I,NF(I)
125     FORMAT(5X,I3,18X,I5)
54    CONTINUE
C
      DO 971 I=1,NTIME
       WRITE (UNIT=8,FMT=301) TSYSR(I)
301    FORMAT (5X,F10.2)
       WRITE (UNIT=4,FMT=341) TSYSF(I)
341    FORMAT (5X,F10.2)
971   CONTINUE
C
      WRITE (6,371)
371   FORMAT (/////,5X,'SIMULATION IS NOW COMPLETE',/////)
C
      END
```

SUBROUTINE SYST

```
      SUBROUTINE SYST (SYSF,T,N,KST,TEMP2,NF)
C
C ******************************************************************
C * CALLED BY PROGRAM MAIN                                         *
C * RETURNS THE SYSTEM FAILURE TIME (SYSF) AND NUMBER OF           *
C * FAILURES PER COMPONENT                                         *
C *                                                                *
C * N      - NUMBER OF COMPONENTS IN THE SYSTEM                    *
C * T      - VECTOR OF COMPONENT FAILURE TIMES (FROM PROGRAM MAIN) *
C * ISYSUP - SYSTEM FUNCTION, MAY BE LARGER THAN ONE               *
C * KST    - COMPONENT STATUS, '0' IS DOWN, '1' IS UP              *
C * TEMP2  - TEMPORARY STORAGE VECTOR TO FIND THE SYSTEM           *
C *          FAILURE TIME                                          *
C * TEMP1  - TIME AT WHICH THE INTERNAL STATUS OF EACH COMPONENT IS *
C *          CHECKED                                               *
C * NF     - NUMBER OF FAILURES PER COMPONENT                      *
C ******************************************************************
C
      DIMENSION KST(20),TEMP2(20),T(20),NF(20)
      J = 0
C
      DO 20 I = 1,N
        TEMP1 = T(I)
C
C **********************************************
C * CHECK STATUS OF EACH COMPONENT AT TIME TEMP1 *
C * CALCULATE SYSTEM STATUS AT TIME TEMP1        *
C * IF SYSTEM STATUS IS 0, STORE TIME IN TEMP2   *
C **********************************************
C
         DO 30 L = 1,N
            IF (T(L).LT.TEMP1.OR.T(L).EQ.TEMP1) THEN
              KST(L) = 0
            ELSE
              KST(L) = 1
            ENDIF
30       CONTINUE
C
C ***********************************************************
C * ENTER THE SYSTEM FUNCTION HERE                          *
C                                                           *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE          *
C * KST(I) - COMPONENT STATUS, 0 OR 1                       *
C ***********************************************************
C
     ISYSUP = KST(1)*KST(2)*KST(3)*KST(4)
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20    CONTINUE
C
C ********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ********************************************
C
      IF (J.GT.0) THEN
         TEMP3 = TEMP2(1)
         DO 70 I = 1,J
             TEMP4 = TEMP2(I)
             IF (TEMP4.LT.TEMP3) TEMP3 = TEMP4
70       CONTINUE
         SYSF = TEMP3
      ELSE
      WRITE (UNIT=6,FMT=90)
90    FORMAT ('UNABLE TO RETURN SYSTEM FAILURE TIME TO PROGRAM MAIN')
      STOP
```

```
      ENDIF
C
C ************************************************
C * CALCULATE NUMBER OF FAILURES PER COMPONENT NF  *
C ************************************************
C
      DO 200 I = 1,N
      IF (T(I).LT.SYSF.OR.T(I).EQ.SYSF) NF(I) = NF(I)+1
200   CONTINUE
C
      RETURN
      END
```

SUBROUTINE FAILT

```
      SUBROUTINE FAILT (TI,KF,FI,FJ,IC,PT)
C
C  ******************************************************
C  * CALLED BY PROGRAM MAIN                             *
C  * CALLS FUNCTION STBY                                *
C  * CALLS FUNCTION PARL                                *
C  * CALLS SUBROUTINE ETIME                             *
C  *                                                    *
C  * SUBROUTINE CALLED FOR EACH COMPONENT               *
C  * SUBROUTINE RETURNS (TI) FOR EACH COMPONENT         *
C  * THE COMPONENT COULD BE A CAT COMPONENT (ACTIVE OR  *
C  * NON ACTIVE STANDBY)                                *
C  ******************************************************
C
C  *****************************************************
C  * TI = T(J) IN PROGRAM MAIN                         *
C  * KF = KFDN(J) IN PROGRAM MAIN                      *
C  * FI = FPTR(J,1) IN PROGRAM MAIN                    *
C  * FJ = FPTR(J,2) IN PROGRAM MAIN                    *
C  *                                                   *
C  * IC = ICODE(J) IN PROGRAM MAIN                     *
C  * IC = 0    SINGLE COMPONENT                        *
C  * IC = 52   ACTIVE STANDBY 2 OUT OF 5               *
C  * IC = -2   NON ACTIVE STANDBY OF 2 COMPONENTS      *
C  *                                                   *
C  * PT = PT IN PROGRAM MAIN                           *
C  *       ONLY FOR DIMENSION PURPOSES IN PROGRAM MAIN *
C  *       USED FOR CAT COMPONENTS                     *
C  *       THE ARRAY PT(K) IS USED IN THIS SUBROUTINE  *
C  *                                                   *
C  *****************************************************
C
      DIMENSION PT(5)
      IF (IC) 20,22,24
C
C ***********************************************
C * IF IC =0 GOTO 22                            *
C * IF IC <0 GOTO 20                            *
C * IF IC >0 GOTO 24                            *
C *                                             *
C * IC = 0  SINGLE COMPONENT                    *
C * IC = 52 ACTIVE STANDBY 2 OUT OF 5           *
C * IC = -2 NON ACTIVE STANDBY OF 2 COMPONENTS  *
C * MAXIMUM OF 5 LRU'S ALLOWED IN CAT COMPONENT *
C ***********************************************
C
C
C  ************************************************************
C  * ICJ - NUMBER OF COMPONENTS REQUIRED FOR SUBSYSTEM SUCCESS *
C  ************************************************************
C
25    DO 73 K=1,ICI
         CALL ETIME (TI,KF,FI,FJ)
C
C  ***************************************
C  * ETIME RETURNS TI                    *
C  * CALLED ONLY ONCE FOR SINGLE COMPONENT *
C  * CALLED MANY TIMES FOR A CAT COMPONENT *
C  ***************************************
C
73    PT(K)=TI
C
C  ***********************************
C  * PT(K) AN ARRAY FOR CAT COMPONENTS *
C  ***********************************
C
      IF (IC) 30,79,34
C
30    TI=STBY(ICI,PT)
```

```
C
C ****************************************
C * SUBSYSTEM IN STANDBY CONFIGURATION   *
C * CALLS FUNCTION STBY                  *
C ****************************************
C
34     TI=PARL(ICI,ICJ,PT)
C
C *************************************
C * SUBSYSTEM IN ACTIVE PARALLEL      *
C * CALLS FUNCTION PARL               *
C *************************************
C
79     RETURN
       END
```

FUNCTION STBY

```
      FUNCTION STBY (NEL,T)
C
C *******************************************************
C *  CALLED BY SUBROUTINE FAILT                         *
C *                                                     *
C *  RETURNS CAT COMPONENT TIME TO FAIL (STBY) FOR      *
C *  COMPONENTS IN NON ACTIVE STANDBY                   *
C *  NEL =  ICI IN SUBROUTINE FAILT                     *
C *  ICI = -IC  = ICODE IN PROGRAM MAIN)                *
C *  IC  = -2 NON ACTIVE STANDBY OF TWO COMPONENTS ETC  *
C *  T   =  PT  IN SUBROUTINE FAILT                     *
C *  MAXIMUM OF 5 COMPONENTS IN CAT COMPONENT           *
C *******************************************************
C
      DIMENSION T(5)
      DO 1 I=1,NEL
1     STBY=STBY+T(I)
C
C *******************************************************
C * ARRAY FOR T(I) IS KEPT IN SUBROUTINE FAILT AS PT(K) *
C * STBY IS A SUMMATION OF EACH SUBCOMPONENTS LIFE      *
C *******************************************************
C
      RETURN
      END
```

FUNCTION PARL

```
      FUNCTION PARL (NEL,NSUS,PT)
C
C ***************************************************************
C * CALLED BY SUBROUTINE FAILT                                  *
C *                                                             *
C * DETERMINES THE TIME TO FAIL FOR A SUBSYSTEM OF UP TO FIVE   *
C * COMPONENTS IN ACTIVE PARALLEL REDUNDANCY GIVEN RANDOM TIME *
C * TO FAIL VECTOR PT(K) IN SUBROUTINE FAILT                   *
C * NEL  =ICI IN SUBROUTINE FAILT - IS THE NUMBER OF COMPONENTS*
C * NSUS =ICJ IN SUBROUTINE FAILT                              *
C *       IS THE NUMBER REQUIRED FOR SUBSYSTEM SUCCESS         *
C ***************************************************************
C
      DIMENSION PT (5)
C
C ***********************
C * SORT PT LOW TO HIGH *
C ***********************
C
      DO 15 I=2,NEL
            IF (PT(I)-PT(I-1)) 10,15,15
10          TEMP=PT(I)
            IM=I-1
            DO 20 J=1,IM
                  L=I-J
13                PT(L+1)=PT(L)
20          CONTINUE
            PT(1)=TEMP
            GOTO 15
14          PT(L+1)=TEMP
15    CONTINUE
C
C  *************************************************
C  * TIMES TO FAIL ARE ODERED FROM LOWEST TO HIGHEST *
C  * EG FOR 2 OUT OF 5 WE SELECT THE TIME TO FAIL OF *
C  * THE 5 - 2 + 1 = 4TH COMPONENT                   *
C  *************************************************
C
      PARL=PT(NO)
C
      RETURN
      END
```

SUBROUTINE ETIME

```
      SUBROUTINE ETIME (TI,ID,P1,P2)
C
C ************************************************************
C * CALLED BY SUBROUTINE FAILT FOR TIME TO FAILURE           *
C * CALLED BY PROGRAM MAIN FOR REPAIRS                       *
C * CALLS SUBROUTINE RAND                                    *
C * R IS BETWEEN 0 AND 1                                     *
C * SUBROUT'NE DETERMINES A RANDOM TIME FROM DISTRIBUTION ID *
C * TO FAIL OR REPAIR DEPENDING ON P1 AND P2                 *
C *                                                          *
C * TI - RANDOM TIME TO FAIL RETURNED TO FAILT AND           *
C *      CAT COMPONENT TIME TO FAIL CALCULATED BY FAILT      *
C *      RDUM,RANDOM REPAIR TIME RETURNED TO MAIN            *
C * ID - KF,KFDN  1,2,3 OR 4                                 *
C *      KRDN                                                *
C * P1 - FI,FPTR(1)  COMPULSORY FOR ALL DISTRIBUTIONS        *
C *      RPTR(1)                                             *
C * P2 - FJ,FPTR(2)  OPTIONAL,DEPENDING ON THE               *
C *      RPTR(2)     DITRIBUTION                             *
C ************************************************************
C
      K=ID
C
C ********************************************
C * EXPONENTIAL DISTRIBUTION, WHERE P1=MTBF  *
C * TI IS ALWAYS POSITIVE                    *
C ********************************************
C
1     CALL RAND (R)
      TI=-P1*(ALOG(R))
      RETURN
C
C **********************************************************
C * NORMAL DISTRIBUTION WITH MEAN P1, STANDARD DEVIATION P2 *
C * TI IS ALWAYS POCITIVE                                   *
C **********************************************************
C
2     CALL RAND (RA)
      CALL RAND (RB)
      V=(-2.*ALOG(RA))**.5*COS(6.2834*RB)
      TI=V*P2+P1
      IF (TI) 19,20,20
19    TI=0
21    TI=EXP(TI)
22    RETURN
C
C ************************************************************
C * WEIBULL DISTRIBUTION, R(T)=EXP(-(T/A)**B) WHERE P1=A, P2=B *
C * A - SCALE PARAMETER A=CHARACTERISTIC LIFE                *
C *     THE CHARACTERISTIC LIFE IS THE TIME AT WHICH 63.2 % OF *
C *     ITEMS HAVE FAILED                                    *
C * B - SHAPE PARAMETER                                      *
C *     B < 1 DECREASING HAZARD RATE                         *
C *     B=1 SAME AS EXPONENTIAL,CONSTANT HAZARD RATE         *
C *     B > 1 INCREASING HAZARD RATE                         *
C *     B=3.5 OR HIGHER APPROXIMATES NORMAL DISTRIBUTION     *
C * TI IS ALWAYS POSITIVE                                    *
C ************************************************************
C
3     CALL RAND (R)
      TI=P1*(-ALOG(R))**P
      RETURN
C
C ************************************************************
C * LOGNORMAL DISTRIBUTION WITH MEAN P1, STANDARD DEVIATION P2 *
C * A VARIABLE WHOSE LOGARITHM FOLLOWS THE NORMAL PROBABILITY  *
C * LAW                                                      *
C * VARIABLE X, Y = LN X IS NORMALLY DISTRIBUTED, MEAN       *
C * AND VARIANCE RELATE TO Y                                 *
```

```
C * P1 - MEAN WHICH RELATES TO THE NATURAL LOG OF THE VARIABLE *
C * P2 - STD DEVIATION WHICH RELATES TO THE NATURAL LOG OF THE *
C *      VARIABLE                                               *
C *                                                             *
C ***************************************************************
C
C
      RETURN
      END
```

```
C * P1 - MEAN WHICH RELATES TO THE NATURAL LOG OF THE VARIABLE *
C * P2 - STD DEVIATION WHICH RELATES TO THE NATURAL LOG OF THE *
C *      VARIABLE                                              *
C *                                                            *
C **************************************************************
C
C
      RETURN
      END
```

SUBROUTINE RAND

```
      SUBROUTINE RAND (Z)
C
C ****************************************************
C * COMBINED CONGRUENTIAL RANDOM NUMBER GENERATOR *
C * FOR SMALL WORD SIZE AND LONG CYCLE LENGTH     *
C ****************************************************
C
      CALL RAND1 (W)
      CALL RAND2 (X)
      CALL RAND3 (Y)
C
      Z = AMOD (W+X+Y,1)
C
      RETURN
      END
```

SUBROUTINE RAND1

```
      SUBROUTINE RAND1 (Z)
C
      DOUBLE PRECISION DM,DSEED
      DATA DM /30269/
      DATA DSEED /5/
C
      DSEED = DMOD (171*DSEED,DM)
      RETURN
      END
```

SUBROUTINE RAND2

```
      SUBROUTINE RAND2 (Z)
C
      DOUBLE PRECISION DM,DSEED
      DATA DM /30307/
      DATA DSEED /11/
C
      DSEED = DMOD (172*DSEED,DM)
      RETURN
      END
```

SUBROUTINE RAND3

```
      SUBROUTINE RAND3 (Z)
C
      DOUBLE PRECISION DM,DSEED
      DATA DM /30323/
      DATA DSEED /7/
C
      DSEED = DMOD (170*DSEED,DM)
      RETURN
      END
```

SUBROUTINE TAB

```
      SUBROUTINE TAB(A,FI,NT,KFREQ,ISZE,MS)
C
C *************************************************************
C * CALLED BY PROGRAM MAIN                                      *
C * A  - TSYSF(I) IN MAIN                                       *
C * NT - SAMPLE SIZE (NTIME IN MAIN)                            *
C * SORTS DATA INTO FREQUENCY CLACSES OF SIZE FI, DETERMINES    *
C * AND PRINTS RESULTING FREQUENCY AND EMPIRICAL                *
C * PROBABILITY DISTRIBUTIONS                                   *
C * ESTIMATES MEAN AND VARIANCE OF SAMPLES                      *
C *************************************************************
C
      COMMON  RMAXTM,RMINTM
      COMMON  /CRES/ RES,RESQ,NTYPE
C
      DIMENSION KFREQ(2),PN(5),ZN(5)
C
C ******************************************************
C * SORT DATA INTO CLASSES, COMPUTE MEAN AND VARIANCE  *
C ******************************************************
C
      IF (MS) 2,4,2
C
C  ***************************************************************
C  * MS   - CONTROL VARIABLE, ORIGINALLY SET TO 1 IN PROGRAM MAIN *
C  *        AND SET AGAIN TO ZERO IN SUBROUTINE TAB               *
C  *        THE PROGRAM ONLY PASSES THROUGH HERE ONCE THEN        *
C  *        TO LINE 4                                             *
C  * ISZE - NUMBER OF FREQUENCY CLASSES                           *
C  * KI   - SAMPLE SIZE                                           *
C  ***************************************************************
C
1     KFREQ(I)=0
      KI=NT
C
C  *******************
C  * POISSON TABLE   *
C  *******************
C
      PN(1)=2.996
      PN(2)=4.744
      PN(3)=6.296
      PN(4)=7.655
      PN(5)=9.155
      ZN(1)=.0516
      ZN(2)=.3530
      ZN(3)=.8168
      ZN(4)=1.3651
      ZN(5)=1.9686
C
      KOVR=0
      SUM=0.
      SUMSQ=0.
C
C
C  ***************************************************************
C  * A  - SYSTEM FAILURE TIME, TSYSF(I) IN PROGRAM MAIN          *
C  * FI - SIZE OF THE CLASSES, FROM PROGRAM MAIN                 *
C  * J  - FREQUENCY CLASS IN WHICH THE SYSTEM FAILURE TIME FALLS *
C  ***************************************************************
C
      IF(J-ISZE) 87,87,86
C
C  ***********************************************************
C  * IF J IS LARGER THEN ISZE THEN GOTO 86 OTHERWISE GOTO 87 *
C  * CALCULATE NUMBER OF OUTLIERS                            *
C  ***********************************************************
C
86    KOVR=KOVR+1
```

```
      GOTO 88
87    KFREQ(J)=KFREQ(J)+1
C
C ******************************************************************
C * KFREQ - FREQUENCY COUNTER, EXCLUDES OVERFLOW NUMBERS          *
C *         KFREQ IS RETURNED TO PROGRAM MAIN                      *
C * KOVR  - OVERFLOW COUNT                                         *
C * ALL SYSTEM FAILURE TIMES ARE USED TO CALCULATE SUM AND SUMSQ *
C ******************************************************************
C
88    SUM=SUM+A
      SUMSQ=SUMSQ+A*A
C
C
C  ******************************************************
C  * KI - SAMPLE SIZE AND 1 IS SUBTRACTED EVERY TIME THE *
C  *      PROGRAM PASSES THROUGH HERE                    *
C  * THE PROGRAM ONLY PASSES THROUGH HERE FOR THE LAST   *
C  * HISTORY (KFREQ,SUM,SUMSQ HAVE BEEN CALCULATED)      *
C  ******************************************************
C
      IF(KI) 15,5,15
5     ANT=NT
C
      VAR=(SUMSQ-ANT*TMEAN*TMEAN)/(ANT-1.)
      SIGMA=SQRT(VAR)
C
C *******************************************
C * TMEAN - MEAN OF SYSTEM FAILURE TIMES     *
C * VAR   - VARIANCE OF SYSTEM FAILURE TIMES *
C *******************************************
C
      IF(NTYPE.EQ.0) GOTO 200
C
C ******************************************************
C ******************************************************
C ******************************************************
C * ONLY FOR AVAILABILITY CALCULATION                   *
C ******************************************************
C
      WRITE(UNIT=3,FMT=96)
96    FORMAT(/,2X,18HAVAILABILITY BLOCK)
      RBAR=RES/AN
      RVAR=(RESQ-AN*RBAR*RBAR)/(AN-1.)
      SIGM=SQRT(RVAR)
C
C ****************************************
C * RBAR = MEAN OF SYSTEM REPAIR TIMES     *
C * RVAR = VARIANCE OF SYSTEM REPAIR TIMES *
C ****************************************
C
      VARN=(VAR/(TMEAN*TMEAN)+RVAR/(RBAR*RBAR))/(16.*AN)
      IN=90
      IT=95
      SIG=SQRT(VARN)
      AV1=1.284*SIG+.5
      CON1=(1.-AV1)*AVAL/(AV1-2.*AV1*AVAL+AVAL)
      AV2=1.645*SIG+.5
      CON2=(1.-AV2)*AVAL/(AV2-2.*AV2*AVAL+AVAL)
C
      WRITE(UNIT=3,FMT=97) TMEAN,SIGMA
97    FORMAT(/,2X,17HAVERAGE UPTIME...,F10.2,
     *16H     SIGMA......,F10.2)
C
      WRITE(UNIT=3,FMT=98) RBAR,SIGM
98    FORMAT(2X,17HAVERAGE DNTIME...,F10.2,
     *16H     SIGMA......,F10.2,/)
C
      WRITE(UNIT=3,FMT=348) AVAL,SIG
348   FORMAT(2X,22HAVERAGE AVAILABILITY..,F5.2,
     *16H     SIGMA......,F10.2,/)
```

```
C
      WRITE(UNIT=3,FMT=99) IN,CON1
      WRITE(UNIT=3,FMT=99) IT,CON2
99    FORMAT(2X,3HTHE,I3,
     *28H PERCENT CONFIDENCE LEVEL...,F6.2)
C
200   CONTINUE
C
C ****************************************************************
C ****************************************************************
C ****************************************************************
C * FIND EMPIRICAL RELIABILITY PROBABILITY DISTRIBUTION          *
C ****************************************************************
C
      SUM2=1.
C
      WRITE(UNIT=3,FMT=106)
106   FORMAT(/,2X,'RELIABILITY BLOCK')
C
      WRITE(UNIT=3,FMT=103) TMEAN,SIGMA
103   FORMAT(/,2X,26HMEAN LIFE.................,F10.2,/,2X,
     *26HSIGMA.....................,F10.2)
C
      WRITE(UNIT=3,FMT=104)
104   FORMAT(/,2X,24HRELIABILITY DISTRIBUTION,//,7X,
     *4HTIME,5X,9HFREQUENCY,3X,4HR95L,6X,4HRMLE,/)
C
      DO 3 I=1,ISZE
        FREQ=KFREQ(I)
        PROB=FREQ/ANT
        SUM2=ABS(SUM2-PROB)
C
C ************************************************************
C * KFREQ - NUMBER OF FAILURES IN EACH INTERVAL              *
C * SUM2  - CUMULATIVE PROBABILITY AT THE END OF EACH        *
C *         INTERVAL AFTER SUBTRACTING THE PROBABLITIES      *
C *         FOR EACH INTERVAL                                *
C ************************************************************
C
        ICUM=ANT*SUM2
C
C ************************************************************
C * ICUM - CUMULATIVE NUMBER OF SURVIVALS UP TO THIS POINT   *
C * IN TIME GIVEN BY SCALE=SCALE+FI                          *
C ************************************************************
C
C ****************************************
C * 95 % LOWER CONFIDENCE LIMIT (NORMAL) *
C ****************************************
C
        RL=SUM2-1.645*SQRT(SUM2*(1.-SUM2)/ANT)
C
C ************************************************************
C * 95 % LOWER CONFIDENCE LIMIT (POISSON)                    *
C * USED FOR THE EXTREMES, FIRST 5 AND LAST 5 FAILURES       *
C ************************************************************
C
        Z=.05**(1./ANT)
        IF (ICUM.EQ.NT) RL=Z
        IF (ICUM.EQ.0)  RL=0
        IF(ICUM.GT.0.AND.ICUM.LT.6) RL=ZN(ICUM)/ANT
        IN=NT-ICUM
        IF(IN.GT.0.AND.IN.LT.6) RL=1.-PN(IN)/ANT
C
        IF (RL.LT.0.) RL=0.
C
        WRITE(UNIT=3,FMT=101) SCALE,KFREQ(I),RL,SUM2
101     FORMAT(2X,F10.2,6X,I4,3X,F6.3,5X,F6.3)
C
C
3     CONTINUE
```

```
C
      FOVR=KOVR
C
C ************************************************************
C * FOVR - OVERFLOW (ISZE AND FI ARE SELECTED BY USER)  *
C ************************************************************
C
      PROB=FOVR/ANT
      SUM2=SUM2-PROB
C
C ****************************************************************
C * THIS IS A CHECK, SUM2 BEING THE CUMULATIVE PROBABILITY  *
C * AT THE END OF THE LAST INTERVAL WHILE PROB IS THE       *
C * PROBABILITY OF THE OVERFLOW, SUBTRACT THE TWO           *
C * AND WE SHOULD HAVE ZERO                                 *
C ****************************************************************
C
      WRITE(UNIT=3,FMT=105) KOVR,SUM2
105   FORMAT(/,2X,8HOVERFLOW,6X,I4,16X,F6.3)
C
C
      WRITE (UNIT=3,FMT=522) ISZE,FI,RMAXTM,RMINTM
522   FORMAT (/,2X,30HNUMBER OF CLASS INTERVALS.....,I7,/,
     *2X,30HCLASS INTERVAL WIDTH...........,F10.2,/,
     *2X,30HMAXIMUM SYSTEM FAILURE TIME...,F10.2,/,
     *2X,30HMINIMUM SYSTEM FAILURE TIME...,F10.2)
C
15    RETURN
      END
```

APPENDIX B

EWS RELIABILITY SIMULATION

RAMIN – EWS

```
NTYPE...0NTIME...5000N...11KFIX... 1IPROB... 1ISZE...12FI... 50.0      END
ICODE...0    LRU NO.1
KFDN... 1FPTR(1)...  .8000000E+03FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.2
KFDN... 1FPTR(1)...  .2500000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .2500000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .1000000E+05FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.5
KFDN... 1FPTR(1)...  .1000000E+05FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.6
KFDN... 1FPTR(1)...  .1000000E+05FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.7
KFDN... 1FPTR(1)...  .1000000E+05FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.8
KFDN... 1FPTR(1)...  .1000000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...21   LRU NO.9
KFDN... 1FPTR(1)...  .3440000E+03FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...21   LRU NO.10
KFDN... 1FPTR(1)...  .3440000E+03FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...32   LRU NO.11
KFDN... 1FPTR(1)...  .6190000E+03FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST – EWS

```
      SUBROUTINE SYST (SYSF,T,N,KST,TEMP2,NF)
C
C ********************************************************************
C * CALLED BY PROGRAM MAIN                                            *
C * RETURNS THE SYSTEM FAILURE TIME (SYSF) AND NUMBER OF              *
C * FAILURES PER COMPONENT                                            *
C *                                                                  *
C * N      - NUMBER OF COMPONENTS IN THE SYSTEM                        *
C * T      - VECTOR OF COMPONENT FAILURE TIMES (FROM PROGRAM MAIN)    *
C * ISYSUP - SYSTEM FUNCTION, MAY BE LARGER THAN ONE                  *
C * KST    - COMPONENT STATUS, '0' IS DOWN, '1' IS UP                 *
C * TEMP2  - TEMPORARY STORAGE VECTOR TO FIND THE SYSTEM              *
C *          FAILURE TIME                                             *
C * TEMP1  - TIME AT WHICH THE INTERNAL STATUS OF EACH COMPONENT IS  *
C *          CHECKED                                                  *
C * NF     - NUMBER OF FAILURES PER COMPONENT                         *
C ********************************************************************
C
      DIMENSION KST(20),TEMP2(20),T(20),NF(20)
      J = 0
C
      DO 20 I = 1,N
        TEMP1 = T(I)
C
C ************************************************
C * CHECK STATUS OF EACH COMPONENT AT TIME TEMP1 *
C * CALCULATE SYSTEM STATUS AT TIME TEMP1        *
C * IF SYSTEM STATUS IS 0, STORE TIME IN TEMP2   *
C ************************************************
C
         DO 30 L = 1,N
            IF (T(L).LT.TEMP1.OR.T(L).EQ.TEMP1) THEN
              KST(L) = 0
            ELSE
              KST(L) = 1
            ENDIF
30       CONTINUE

C
C ***********************************************************
C * ENTER THE SYSTEM FUNCTION HERE                          *
C *                                                         *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE          *
C * KST(I) - COMPONENT STATUS, 0 OR 1                       *
C ***********************************************************
C
     K1=KST(1)*KST(2)*KST(4)*KST(6)*KST(8)*KST(9)*KST(10)*KST(11)
     K2=KST(1)*KST(3)*KST(5)*KST(7)*KST(8)*KST(9)*KST(10)*KST(11)
     ISYSUP=K1+K2
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20    CONTINUE
C
C ********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ********************************************
C
      IF (J.GT.0) THEN
         TEMP3 = TEMP2(1)
         DO 70 I = 1,J
              TEMP4 = TEMP2(I)
              IF (TEMP4.LT.TEMP3) TEMP3 = TEMP4
70       CONTINUE
         SYSF = TEMP3
      ELSE
```

```
      WRITE (UNIT=6,FMT=90)
90    FORMAT ('UNABLE TO RETURN SYSTEM FAILURE TIME TO PROGRAM MAIN')
      STOP
      ENDIF
C
C ************************************************
C * CALCULATE NUMBER OF FAILURES PER COMPONENT NF  *
C ************************************************
C
      DO 200 I = 1,N
      IF (T(I).LT.SYSF.OR.T(I).EQ.SYSF) NF(I) = NF(I)+1
200   CONTINUE
C
      RETURN
      END
```

RAMOUT – EWS

```
                              RUN NO.  1

                             INPUT BLOCK

TYPE OF RUN (0,1,OR 2)............................         0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM.................       11
REPAIR SPECIFICATION (1,2 OR 3)....................        1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       12
CLASS INTERVAL WIDTH...............................    50.00

COMP NO. 1 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8000000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2500000E+04  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2500000E+04  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+05  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+05  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+05  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+05  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+04  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 9 ICODE..21
           KFDN.. 1  FPTR(1)..  .3440000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.10 ICODE..21
           KFDN.. 1  FPTR(1)..  .3440000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.11 ICODE..32
           KFDN.. 1  FPTR(1)..  .6190000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

                              OUTPUT BLOCK


RELIABILITY BLOCK

MEAN LIFE................     169.05
SIGMA....................     127.20

RELIABILITY DISTRIBUTION

     TIME      FREQUENCY   R95L      RMLE

     50.00        785      .835      .843
```

100.00	959	.640	.651
150.00	877	.464	.476
200.00	769	.311	.322
250.00	541	.204	.214
300.00	347	.136	.144
350.00	259	.086	.093
400.00	171	.053	.058
450.00	103	.033	.038
500.00	77	.019	.022
550.00	46	.011	.013
600.00	22	.007	.009
OVERFLOW	44		.000

NUMBER OF CLASS INTERVALS..... 12
CLASS INTERVAL WIDTH.......... 50.00
MAXIMUM SYSTEM FAILURE TIME... 921.03
MINIMUM SYSTEM FAILURE TIME... .07

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	1050
2	331
3	310
4	73
5	89
6	91
7	70
8	881
9	1000
10	1033
11	986

APPENDIX C

SHUTTLE COMPUTER SYSTEM RELIABILITY SIMULATION

RAMIN - CONFIG 1

```
NTYPE...ONTIME...5000H... 4KFIX... 1IPROB... 1ISZE... 8FI...    1.0   END
ICODE...0    LRU NO.1
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.2
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+01RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 1

```
      SUBROUTINE SYST (SYSF,T,N,KST,TEMP2,NF)
C
C ******************************************************************
C * CALLED BY PROGRAM MAIN                                         *
C * RETURNS THE SYSTEM FAILURE TIME (SYSF) AND NUMBER OF           *
C * FAILURES PER COMPONENT                                         *
C *                                                                *
C * N      - NUMBER OF COMPONENTS IN THE SYSTEM                    *
C * T      - VECTOR OF COMPONENT FAILURE TIMES (FROM PROGRAM MAIN) *
C * ISYSUP - SYSTEM FUNCTION, MAY BE LARGER THAN ONE               *
C * KST    - COMPONENT STATUS, '0' IS DOWN, '1' IS UP              *
C * TEMP2  - TEMPORARY STORAGE VECTOR TO FIND THE SYSTEM           *
C *          FAILURE TIME                                          *
C * TEMP1  - TIME AT WHICH THE INTERNAL STATUS OF EACH COMPONENT IS *
C *          CHECKED                                               *
C * NF     - NUMBER OF FAILURES PER COMPONENT                      *
C ******************************************************************
C
      DIMENSION KST(20),TEMP2(20),T(20),NF(20)
      J = 0
C
      DO 20 I = 1,N
        TEMP1 = T(I)
C
C ********************************************
C * CHECK STATUS OF EACH COMPONENT AT TIME TEMP1 *
C * CALCULATE SYSTEM STATUS AT TIME TEMP1        *
C * IF SYSTEM STATUS IS 0, STORE TIME IN TEMP2   *
C ********************************************
C
         DO 30 L = 1,N
            IF (T(L).LT.TEMP1.OR.T(L).EQ.TEMP1) THEN
              KST(L) = 0
            ELSE
              KST(L) = 1
            ENDIF
30       CONTINUE
C
C ************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                           *
C *                                                          *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE           *
C * KST(I) - COMPONENT STATUS, 0 OR 1                        *
C ************************************************************
C
      ISYSUP = KST(1)*KST(2)*KST(3)*KST(4)
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20    CONTINUE
C
C ********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ********************************************
C
      IF (J.GT.0) THEN
         TEMP3 = TEMP2(1)
         DO 70 I = 1,J
             TEMP4 = TEMP2(I)
             IF (TEMP4.LT.TEMP3) TEMP3 = TEMP4
70       CONTINUE
         SYSF = TEMP3
      ELSE
      WRITE (UNIT=6,FMT=90)
90    FORMAT ('UNABLE TO RETURN SYSTEM FAILURE TIME TO PROGRAM MAIN')
      STOP
```

```
      ENDIF
C
C ***********************************************
C * CALCULATE NUMBER OF FAILURES PER COMPONENT NF  *
C ***********************************************
C
      DO 200 I = 1,N
      IF (T(I).LT.SYSF.OR.Y(I).EQ.SYSF) NF(I) = NF(I)+1
200   CONTINUE
C
      RETURN
      END
```

RAMOUT - CONFIG 1

RUN NO. 1

INPUT BLOCK

TYPE OF RUN (0,1,OR 2)............................	0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......	5000
NUMBER OF COMPONENTS IN THE SYSTEM................	4
REPAIR SPECIFICATION (1,2 OR 3)...................	1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....	8
CLASS INTERVAL WIDTH..............................	1.00

COMP NO. 1 ICODE.. 0

KFDN.. 1	FPTR(1)..	.7500000E+01	FPTR(2)..	.0000000E+00
KRDN.. 0	RPTR(1)..	.0000000E+00	RPTR(2)..	.0000000E+00

COMP NO. 2 ICODE.. 0

KFDN.. 2	FPTR(1)..	.6000000E+01	FPTR(2)..	.1500000E+01
KRDN.. 0	RPTR(1)..	.0000000E+00	RPTR(2)..	.0000000E+00

COMP NO. 3 ICODE.. 0

KFDN.. 1	FPTR(1)..	.1000000E+02	FPTR(2)..	.0000000E+00
KRDN.. 0	RPTR(1)..	.0000000E+00	RPTR(2)..	.0000000E+00

COMP NO. 4 ICODE.. 0

KFDN.. 1	FPTR(1)..	.8400000E+01	FPTR(2)..	.0000000E+00
KRDN.. 0	RPTR(1)..	.0000000E+00	RPTR(2)..	.0000000E+00

OUTPUT BLOCK

RELIABILITY BLOCK

MEAN LIFE.................	2.44
SIGMA.....................	1.94

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	1504	.689	.699
2.00	1013	.485	.497
3.00	771	.331	.342
4.00	582	.216	.226
5.00	482	.122	.130
6.00	361	.052	.057
7.00	196	.015	.018
8.00	71	.003	.004
OVERFLOW	20		.000

NUMBER OF CLASS INTERVALS.....	8
CLASS INTERVAL WIDTH..........	1.00
MAXIMUM SYSTEM FAILURE TIME...	10.28
MINIMUM SYSTEM FAILURE TIME...	.00

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	1659
2	676
3	1189
4	1476

RAMIN - CONFIG 2

```
NTYPE...0NTIME...5000N... 6KFIX... 1IPROB... 2ISZE...10FI...    1.0   END
ICODE...0    LRU NO.1
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.2
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...21   LRU NO.5
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...21   LRU NO.6
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 2

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ****************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                                *
C *                                                              *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE               *
C * KST(I) - COMPONENT STATUS, 0 OR 1                            *
C ****************************************************************
C
      ISYSUP = KST(1)*KST(3)*KST(5)*KST(6)+
     @KST(2)*KST(4)*KST(5)*KST(6)
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20     CONTINUE
C
C *********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C *********************************************
C
```

RAMOUT - CONFIG 2

RUN NO. 2

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2).............................      0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......   5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      6
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....     10
CLASS INTERVAL WIDTH...............................      1.00

COMP NO. 1 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+02  Fr..(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 5 ICODE..21
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 6 ICODE..21
           KFDN.. 1  FPTR(1)..  .7500000E+01          .  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  P'     ..  .0000000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE................      4.25
SIGMA....................      2.12
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	254	.944	.949
2.00	638	.813	.822
3.00	745	.662	.673
4.00	692	.523	.534
5.00	716	.380	.391
6.00	768	.228	.237
7.00	659	.098	.106
8.00	360	.029	.034
9.00	138	.004	.006
10.00	26	.000	.001
OVERFLOW	4		.000

```
NUMBER OF CLASS INTERVALS.....     10
CLASS INTERVAL WIDTH..........      1.00
MAXIMUM SYSTEM FAILURE TIME...     10.62
MINIMUM SYSTEM FAILURE TIME...       .08
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	2127
2	2133
3	1856
4	1816
5	1307
6	1152

RAMIN - CONFIG 3

```
NTYPE...ONTIME...5000N... 4KFIX... 1IPROB... 3ISZE... 8FI...    1.0   END
ICODE...0     LRU NO.1
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...32    LRU NO.2
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+01RPTR(2)...  .0000000E+00END
ICODE...32    LRU NO.3
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.4
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00PPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST – CONFIG 3

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ***************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                              *
C *                                                             *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE              *
C * KST(I) - COMPONENT STATUS, 0 OR 1                           *
C ***************************************************************
C
      ISYSUP = KST(1)*KST(2)*KST(3)*KST(4)
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20     CONTINUE
C
C ***********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ***********************************************
C
```

RAMOUT - CONFIG 3

RUN NO. 3

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)...............................      0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......   5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      4
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....      8
CLASS INTERVAL WIDTH...............................      1.00

COMP NO. 1 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE..32
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE..32
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .7500000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE.................     2.65
SIGMA.....................     1.90
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	1191	.752	.762
2.00	1042	.542	.553
3.00	859	.370	.382
4.00	617	.248	.258
5.00	552	.140	.148
6.00	437	.055	.060
7.00	240	.010	.012
8.00	52	.001	.002
OVERFLOW	10		.000

```
NUMBER OF CLASS INTERVALS.....     8
CLASS INTERVAL WIDTH..........     1.00
MAXIMUM SYSTEM FAILURE TIME...     9.00
MINIMUM SYSTEM FAILURE TIME...      .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	1593
2	954
3	675
4	1778

RAMIN - CONFIG 4

```
NTYPE...0NTIME...5000N... 4KFIX... 1IPROB... 4ISZE...15FI...    1.0   END
ICODE...0    LRU NO.1
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...-3   LRU NO.2
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+01RPTR(2)...  .0000000E+00END
ICODE...-3   LRU NO.3
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.6
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 4

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ****************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                               *
C *                                                              *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE               *
C * KST(I) - COMPONENT STATUS, 0 OR 1                            *
C ****************************************************************
C
      ISYSUP = KST(1)*KST(2)*KST(3)*KST(4)
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20     CONTINUE
C
C **********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C **********************************************
C
```

RAMOUT - CONFIG 4

RUN NO. 4

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2).............................        0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM.................        4
REPAIR SPECIFICATION (1,2 OR 3)....................        1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       15
CLASS INTERVAL WIDTH...............................        1.00

COMP NO. 1 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE..-3
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE..-3
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .7500000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE.................      3.83
SIGMA.....................      3.60
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	1082	.774	.784
2.00	875	.597	.609
3.00	734	.450	.462
4.00	531	.344	.356
5.00	400	.265	.276
6.00	316	.203	.212
7.00	239	.156	.165
8.00	195	.118	.126
9.00	136	.091	.098
10.00	112	.070	.076
11.00	87	.053	.059
12.00	82	.038	.042
13.00	48	.028	.033
14.00	42	.021	.024
15.00	39	.013	.016
OVERFLOW	82		.000

```
NUMBER OF CLASS INTERVALS.....     15
CLASS INTERVAL WIDTH..........      1.00
MAXIMUM SYSTEM FAILURE TIME...     21.73
MINIMUM SYSTEM FAILURE TIME...       .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	2293

2	113
3	47
4	2547

RAMIN - CONFIG 5

```
NTYPE...ONTIME...5000N... 7KFIX... 1IPROB... 5ISZE... 7FI...    1.0   END
ICODE...31   LRU NO.1
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...31   LRU NO.2
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.5
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.6
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.7
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 5

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ****************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                               *
C *                                                              *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE               *
C * KST(I) - COMPONENT STATUS, 0 OR 1                            *
C ****************************************************************
C
      ISYSUP=KST(1)*KST(2)*KST(3)*KST(4)*KST(5)*KST(6)*KST(7)
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20     CONTINUE
C
C *******************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C *******************************************
C
```

RAMOUT - CONFIG 5

RUN NO. 5

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)..............................         0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......      5000
NUMBER OF COMPONENTS IN THE SYSTEM.................         7
REPAIR SPECIFICATION (1,2 OR 3)....................         1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....         7
CLASS INTERVAL WIDTH...............................         1.00

COMP NO. 1 ICODE..3*
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE..31
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 5 ICODE.. 0
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .7500000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE..................     1.95
SIGMA......................     1.66
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	1869	.615	.626
2.00	1143	.386	.398
3.00	821	.224	.233
4.00	481	.129	.137
5.00	365	.058	.064
6.00	199	.021	.024
7.00	93	.004	.006
OVERFLOW	29		.000

```
NUMBER OF CLASS INTERVALS.....     7
CLASS INTERVAL WIDTH..........     1.00
MAXIMUM SYSTEM FAILURE TIME...     8.25
MINIMUM SYSTEM FAILURE TIME...      .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	66
2	79
3	1169
4	946
5	320
6	1155
7	1265

RAMIN - CONFIG 6

```
NTYPE...0NTIME...5000N... 7KFIX... 1IPROB... 6ISZE... 7FI...    1.0   END
ICODE...-3   LRU NO.1
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...-3   LRU NO.2
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .84000[illegible]01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .00000[illegible]0RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.5
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.6
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.7
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 6

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ***************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                              *
C *                                                             *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE              *
C * KST(I) - COMPONENT STATUS, 0 OR 1                           *
C ***************************************************************
C
      ISYSUP=KST(1)*KST(2)*KST(3)*KST(4)*KST(5)*KST(6)*KST(7)
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20     CONTINUE
C
C ********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ********************************************
C
```

RAMOUT – CONFIG 6

RUN NO. 6

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)..............................     0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......   5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      7
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....      7
CLASS INTERVAL WIDTH...............................      1.00

COMP NO. 1 ICODE..-3
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE..-3
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1000000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 5 ICODE.. 0
           KFDN.. 2  FPTR(1)..  .6000000E+01  FPTR(2)..  .1500000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .8400000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .7500000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE.................      1.98
SIGMA.....................      1.72
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1.00	1868	.615	.626
2.00	1138	.387	.399
3.00	803	.228	.238
4.00	484	.133	.141
5.00	356	.064	.070
6.00	192	.028	.032
7.00	106	.008	.011
OVERFLOW	53		.000

```
NUMBER OF CLASS INTERVALS.....    7
CLASS INTERVAL WIDTH..........    1.00
MAXIMUM SYSTEM FAILURE TIME...    9.80
MINIMUM SYSTEM FAILURE TIME...     .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	0
2	14
3	1182
4	963
5	387
6	1173
7	1281

RAMIN - CONFIG 7

```
NTYPE...0NTIME...5000N... 4KFIX... 1IPROB... 7ISZE... 8FI...    1.0   END
ICODE...42   LRU NO.1
KFDN... 1FPTR(1)...  .1000000E+02FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...42   LRU NO.2
KFDN... 2FPTR(1)...  .6000000E+01FPTR(2)...  .1500000E+01END
KRDN... 0RPTR(1)...  .0000000E+01RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .8400000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .7500000E+01FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - CONFIG 7

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ****************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                               *
C *                                                              *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE               *
C * KST(I) - COMPONENT STATUS, 0 OR 1                            *
C ****************************************************************
C
      ISYSUP = KST(1)*KST(2)*KST(3)*KST(4)
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20     CONTINUE
C
C ***************************************************
C * FIND SMALLEST SYSTEM TIME TO [illegible] SYSF *
C ***************************************************
C
```

RAMOUT - CONFIG 7

```
                              RUN NO.  7

                              INPUT BLOCK

TYPE OF RUN (0,1,OR 2)..............................        0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......      5000
NUMBER OF COMPONENTS IN THE SYSTEM.................         4
REPAIR SPECIFICATION (1,2 OR 3)....................         1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....         8
CLASS INTERVAL WIDTH...............................      1.00

COMP NO. 1 ICODE..42
           KFDN.. 1  FPTR(1)..  .1600000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE..42
           KFDN.. 2  FPTR(1)..  [illegible]000000E+01  FPTR(2)..  .15[illegible]000E+01
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .84[illegible]000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .000[illegible]000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .75[illegible]000E+01  FPTR(2)..  .0000000E+00
           KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00


                              OUTPUT BLOCK

RELIABILITY BLOCK

MEAN LIFE.................     2.95
SIGMA.....................     2.13

RELIABILITY DISTRIBUTION

   TIME     FREQUENCY    R95L      RMLE

   1.00       1171       .756      .766
   2.00        899       .575      .586
   3.00        761       .422      .434
   4.00        617       .300      .310
   5.00        464       .208      .218
   6.00        492       .112      .119
   7.00        445       .026      .030
   8.00        128       .003      .005

OVERFLOW       23                  .000

NUMBER OF CLASS INTERVALS.....     8
CLASS INTERVAL WIDTH..........     1.00
MAXIMUM SYSTEM FAILURE TIME...     9.07
MINIMUM SYSTEM FAILURE TIME...     .00


FAILURE BLOCK

COMPONENT NO.      NUMBER OF FAILURES

     1                    470
     2                    733
     3                   1794
     4                   2003
```

B 747 ELECTRICAL SYSTEM RELIABILITY SIMULATION

RAMIN – B747 ELECTRICAL SYSTEM

```
NTYPE...0NTIME...5000N...15KFIX... 1IPROB... 1ISZE...12FI... 1000.00  END
ICODE...0     LRU NO.1
KFDN... 1FPTR(1)...  .1920000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.2
KFDN... 1FPTR(1)...  .1920000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.3
KFDN... 1FPTR(1)...  .1920000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.4
KFDN... 1FPTR(1)...  .1920000E+04FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.5
KFDN... 1FPTR(1)...  .2000000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.6
KFDN... 1FPTR(1)...  .2000000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.7
KFDN... 1FPTR(1)...  .2000000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.8
KFDN... 1FPTR(1)...  .1111110E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.9
KFDN... 1FPTR(1)...  .1250000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.10
KFDN... 1FPTR(1)...  .2500000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.11
KFDN... 1FPTR(1)...  .2500000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.12
KFDN... 1FPTR(1)...  .2500000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.13
KFDN... 1FPTR(1)...  .4[illegible]1000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.14
KFDN... 1FPTR(1)...  .5000000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
ICODE...0     LRU NO.15
KFDN... 1FPTR(1)...  .5000000E+06FPTR(2)...  .0000000E+00END
KRDN... 0RPTR(1)...  .0000000E+00RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST - B747 ELECTRICAL SYSTEM

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ****************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                                *
C *                                                               *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE                *
C * KST(I) - COMPONENT STATUS, 0 OR 1                             *
C ****************************************************************
C
      K1 = KST(3)*KST(7)*KST(9)*KST(13)*KST(15)+
     @KST(2)*KST(6)*KST(8)*KST(9)*KST(13)*KST(15)+
     @KST(1)*KST(5)*KST(8)*KST(9)*KST(13)*KST(15)
C
      K2 = KST(4)*KST(13)*KST(15)+
     @KST(1)*KST(10)*KST(15)+
     @KST(2)*KST(11)*KST(14)+
     @KST(3)*KST(12)*KST(14)
C
      ISYSUP = K1+K2
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20    CONTINUE
C
C ***********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ***********************************************
C
```

RAMOUT - B747 ELECTRICAL SYSTEM

```
                         RUN NO.  1

                        INPUT BLOCK

TYPE OF RUN (0,1,OR 2)...........................          0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM................        15
REPAIR SPECIFICATION (1,2 OR 3)...................         1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       12
CLASS INTERVAL WIDTH..............................   1000.00

COMP NO. 1 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1920000E+04  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 2 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1920000E+04  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 3 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1920000E+04  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 4 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1920000E+04  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 5 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2000000E+06  FPTR(2)..  .0000000E+00
          KPDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 6 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2000000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 7 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2000000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 8 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1111110E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO. 9 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1250000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.10 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2500000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.11 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2500000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.12 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2500000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.13 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2500000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO.14 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .5000000E+06  FPTR(2)..  .0000000E+00
          KRDN.. 0  RPTR(1)..  .0000000E+00  RPTR(2)..  .0000000E+00

COMP NO 15 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .5000000E+06  FPTR(2)..  .0000000E+00
```

KRDN.. 0 RPTR(1).. .0000000E+00 RPTR(2).. .0000000E+00

OUTPUT BLOCK

RELIABILITY BLOCK

MEAN LIFE.................. 3959.30
SIGMA...................... 2254.38

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
1000.00	136	.969	.973
2000.00	750	.814	.823
3000.00	1088	.594	.605
4000.00	999	.394	.405
5000.00	772	.241	.251
6000.00	460	.150	.159
7000.00	309	.090	.097
8000.00	190	.054	.059
9000.00	115	.032	.036
10000.00	80	.017	.020
11000.00	39	.010	.012
12000.00	30	.005	.006
OVERFLOW	32		.000

NUMBER OF CLASS INTERVALS..... 12
CLASS INTERVAL WIDTH.......... 1000.00
MAXIMUM SYSTEM FAILURE TIME... 18123.66
MINIMUM SYSTEM FAILURE TIME... 228.46

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	4989
2	4991
3	4997
4	4972
5	93
6	113
7	100
8	206
9	170
10	71
11	83
12	79
13	81
14	42
15	42

AMIR® SIMULATION – B747 ELECTRICAL SYSTEM

IRB1 – B747 ELECTRICAL SYSTEM

```
(*      TITLE CARD      *)
1 B747 ELECTRICAL SYSTEM
(*      SYSTEM NAME             TMAX    # OF COMP. # OF TYPE NAMES  *)
          115VAC          2.000000E+04     15     6
(* ATTACHMENT OF COMP. TO THE TYPE NAMES *)
   1    1    1    1    2    2    2    3    4    5    5    5
   5    6    6
(* TYPE NAMES  *)
                         A                         B                         C
                         D                         E                         F
(*  FAILURE RATES  BY TYPE NAMES  *)
 5.208000E-04  5.000000E-06  9.000000E-06  8.000000E-06  4.000000E-06
 2.000000E-06
(*  REPAIR RATES BY TYPE NAMES *)
 1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
 1.000000E+00
(*  FLAG OF PASS.F.R. AND RATES IF FLAG=1 *)
  0
(*  PASSIVE STAND BY PER COMPONENT  *)
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0
(* IDUM ARRAY *)
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
(* RDUM ARRAY *)
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
(* NO. OF LOGICAL STAND BY UNITS    *)
           0
(* NO. OF LOAD CONNECTIONS GROUPS  *)
           0
(* NO. OF INDUCE FAILURES *)
           0
(* REPAIR TEAMS FLAG ;   NO. OF TEAMS  ;  WISER/FIFO POLICY *)
           0              0              0
(*         NPS    (I,J  JSTOP)        NDUMP      PRSD     *)
          5000             1           1000  1.000000E+01
(*         FCB        FCNUM        *)
           0  1.000000E+00
(*         TRB             NTI       BETA   *)
           0             1000  0.000000E+00
(*  NO. OF SENSITIVITY GROUPS  *)
   0
(*  ROFLAG  NO. OF TIME POINTS  &  FLAG INDICATES THAT POINTS WERE ENTERED *)
           1              20              0
(* IRB2 INPUT AND REPAIR MODE FLAGS  *)
```

LBOUT - B747 ELECTRICAL SYSTEM

```
      FUNCTION NSYSCM()
C     Enter the number of components for NSYSCM
      NSYSCM =  15
      return
      END
      SUBROUTINE SYSUP(SYS,T,MI1)
C       PARAMETER 50 IS VALID FOR PC VERSION
      COMMON/SYST/ B(50)
C      for user programing delete the next two lines:
C        WRITE(2,'(A)')' SYSTEM FUNCTION IS MISSING!   ABORTED.'
C        STOP
C     once the above two lines were deleted start programing
C     the system function below this line
C     ================================================
C
      K1 = B(3)*B(7)*B(9)*B(13)*B(15)+
     @B(2)*B(6)*B(8)*B(9)*B(13)*B(15)+
     @B(1)*B(5)*B(8)*B(9)*B(13)*B(15)
C
      K2 = B(4)*B(13)*B(15)+
     @B(1)*B(10)*B(15)+
     @B(2)*B(11)*B(14)+
     @B(3)*B(12)*B(14)
C
      K3 = K1+K2
      IF (K3.GT.0) THEN
         SYS=1
      ELSE
         SYS=0
      ENDIF
C
      RETURN
      END
```

LBOUT - B747 ELECTRICAL SYSTEM

```
      FUNCTION NSYSCM()
C     Enter the number of components for NSYSCM
      NSYSCM =  15
      return
      END
      SUBROUTINE SYSUP(SYS,T,NI1)
C       PARAMETER 50 IS VALID FOR PC VERSION
      COMMON/SYST/ B(50)
C      for user programing delete the next two lines:
C         WRITE(2,'(A)')' SYSTEM FUNCTION IS MISSING!   ABORTED.'
C         STOP
C     once the above two lines were deleted start programing
C     the system function below this line
C     ==============================================
C
      K1 = B(3)*B(7)*B(9)*B(13)*B(15)+
     @B(2)*B(6)*B(8)*B(9)*B(13)*B(15)+
     @B(1)*B(5)*B(8)*B(9)*B(13)*B(15)
C
      K2 = B(4)*B(13)*B(15)+
     @B(1)*B(10)*B(15)+
     @B(2)*B(11)*B(14)+
     @B(3)*B(12)*B(14)
C
      K3 = K1+K2
      IF (K3.GT.0) THEN
         SYS=1
      ELSE
         SYS=0
      ENDIF
C
       RETURN
       END
```

OUTR1 - B747 ELECTRICAL SYSTEM

```
                                                K M I R      VER. 1.2


          ************************************
          *    I N P U T    B L O C K    *
          ************************************

(*      TITLE CARD      *)
1 B747 ELECTRICAL SYSTEM
(*      SYSTEM NAME               TMAX     # OF COMP. # OF TYPE NAMES  *)
             115VAC          2.000000E+04     15      6
(* ATTACHMENT OF COMP. TO THE TYPE NAMES *)
   1    1    1    1    2    2    2    3    4    5    5    5
   5    6    6
(* TYPE NAMES  *)
                          A                         B                        C
                          D                         E                        F
(*  FAILURE RATES  BY TYPE NAMES  *)
 5.208000E-04  5.000000E-06  9.000000E-06  8.000000E-06  4.000000E-06
 2.000000E-06
(*  REPAIR RATES BY TYPE NAMES *)
 1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
 1.000000E+00
(*  FLAG OF PASS.F.R. AND RATES IF FLAG=1 *)
  0
(*  PASSIVE STAND BY PER COMPONENT  *)
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0
(* IDUM ARRAY *)
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
          0           0           0           0           0
(* RDUM ARRAY *)
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
 .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00
(* NO. OF LOGICAL STAND BY UNITS   *)
           0
(* NO. OF LOAD CONNECTIONS GROUPS  *)
           0
(* NO. OF INDUCE FAILURES *)
           0
(* REPAIR TEAMS FLAG ;   NO. OF  TEAMS  ;  WISER/FIFO POLICY *)
           0              0            0
(*        NPS    (I,J  JSTOP)        NDUMP      PRSD      *)
         5000            1           1000  1.000000E+01
(*        FCB       FCNUM          *)
            0  1.000000E+00
(*        TRB            HTI      BETA   *)
            0           1000  0.000000E+00
(*  NO. OF SENSITIVITY GROUPS  *)
   0
(*  ROFLAG  NO. OF TIME POINTS  &  FLAG INDICATES THAT POINTS WERE ENTERED *)
           1            20            0
```

(* IRB2 INPUT AND REPAIR MODE FLAGS *)
0 1

===

B747 ELECTRICAL SYSTEM

RESULTS FOR == 5000 == CASE HISTORIES
AVERAGE NO. OF COLLISIONS PER HISTORY= 2.2741E+01

```
******************************************
*          TIME   DEPENDENT              *
* FAILURE PROBABILITY  (DEMAND MODE)     *
******************************************
```

TIME	FAILURE PROB.	P.R.S.D.
1.0000E+03	2.9600E-02	8.0974E+00
2.0000E+03	1.8540E-01	2.9644E+00
3.0000E+03	4.0380E-01	1.7184E+00
4.0000E+03	6.0420E-01	1.1446E+00
5.0000E+03	7.5380E-01	8.0822E-01
6.0000E+03	8.4540E-01	6.0477E-01
7.0000E+03	9.0320E-01	4.6298E-01
8.0000E+03	9.4120E-01	3.5348E-01
9.0000E+03	9.6080E-01	2.8563E-01
1.0000E+04	9.7560E-01	2.2365E-01
1.1000E+04	9.8460E-01	1.7687E-01
1.2000E+04	9.9040E-01	1.3923E-01
1.3000E+04	9.9420E-01	1.0802E-01
1.4000E+04	9.9720E-01	7.4938E-02
1.5000E+04	9.9820E-01	6.0054E-02
1.6000E+04	9.9900E-01	4.4743E-02
1.7000E+04	9.9940E-01	3.4651E-02
1.8000E+04	9.9980E-01	2.0001E-02
1.9000E+04	1.0000E+00	0.0000E+00
2.0000E+04	1.0000E+00	0.0000E+00

CONDITIONAL MTTF OF THE SYSTEM = 3.93462E+03
THE CONDITIONAL MTTF IS THE REGULAR MTTF ONLY IF
THE UNRELIABILITY AT TMAX IS 1.0

===

```
******************************************
*  S E N S I T I V I T Y   B L O C K *
******************************************
```

C O M P O N E N T SENSITIVITY TABLE

COMPONENT DEPENDENT UNRELIABILITY

1) 4.95E+03	2) 4.99E+03	3) 4.99E+03	4) 4.88E+03	5) 0.00E+00
6) 1.00E+00	7) 1.00E+00	8) 2.00E+00	9) 6.00E+00	10) 2.00E+00
11) 4.00E+00	12) 5.00E+00	13) 2.90E+01	14) 3.00E+00	15) 2.50E+01

NUMBER OF COMPONENT DEPENDENT FAILURES

1) 4948	2) 4987	3) 4986	4) 4881	5) 0
6) 1	7) 1	8) 2	9) 6	10) 2
11) 4	12) 5	13) 29	14) 3	15) 25

NORMALIZED COMPONENT DEPENDENT UNRELIABILITY

```
 1) 9.92E-01    2) 1.00E+00    3) 1.00E+00    4) 9.79E-01    5) 0.00E+00
 6) 2.01E-04    7) 2.01E-04    8) 4.01E-04    9) 1.20E-03   10) 4.01E-04
11) 3.02E-04   12) 1.00E-03   13) 5.82E-03   14) 6.02E-04   15) 5.01E-03
```

```
*******************************************
*   SPARE PARTS AND REPAIR   B L O C K   *
*******************************************
```

REPAIRS IN TIME SURFACES PER COMPONENT

```
 1) 0.00E+00    2) 0.00E+00    3) 0.00E+00    4) 0.00E+00    5) 0.00E+00
 6) 0.00E+00    7) 0.00E+00    8) 0.00E+00    9) 0.00E+00   10) 0.00E+00
11) 0.00E+00   12) 0.00E+00   13) 0.00E+00   14) 0.00E+00   15) 0.00E+00
```

REPAIRS IN TIME SURFACES PER TYPE

```
A )  0.00E+00                         B )  0.00E+00
C )  0.00E+00                         D )  0.00E+00
E )  0.00E+00                         F )  0.00E+00
```

REPAIRS IN CONTINUOUS PROCESS PER COMPONENT

```
 1) 2.36E+00    2) 2.45E+00    3) 2.27E+00    4) 2.26E+00    5) 6.00E-04
 6) 1.20E-03    7) 6.00E-04    8) 3.40E-03    9) 5.49E-03   10) 4.60E-03
11) 7.20E-03   12) 3.80E-03   13) 2.24E-02   14) 5.20E-03   15) 1.96E-02
```

REPAIRS IN CONTINUOUS PROCESS PER TYPE

```
A )  9.34E+00                         B )  2.40E-03
C )  3.40E-03                         D )  5.40E-03
E )  3.80E-02                         F )  2.48E-02
```

MAX. WEIGHT SCORED IN THIS RUN: 1.0000E+00

MIN. WEIGHT SCORED IN THIS RUN: 1.0000E+00

```
===================================================================
```

```
*******************************************
*   D I A G N O S T I C     B L O C K   *
*******************************************
```

===== FIRST SCORE SPECTRUM IN STEPS =====

```
   0      0      0      7  4151    761     76      5
```

===== FIRST DETECTOR SPECTRUM BY STEPS =====

```
0.0000E+00   0.0000E+00   0.0000E+00   7.0000E+00   4.1510E+03
7.6100E+02   7.6000E+01   5.0000E+00
```

```
     NO. OF SCORES :
   148.00     927.00    2019.00    3021.00    3769.00
  4227.00    4516.00    4706.00    4804.00    4878.00
  4923.00    4952.00    4971.00    4986.00    4991.00
  4995.00    4997.00    4999.00    5000.00    5000.00
```

AVERAGE WEIGHT OF COMPONENT DEPENDENT UNRELIABILITY

```
 1) 1.00E+00    2) 1.00E+00    3) 1.00E+00    4) 1.00E+00    5) 0.00E+00
```

6) 1.00E+00	7) 1.00E+00	8) 1.00E+00	9) 1.00E+00	10) 1.00E+00
11) 1.00E+00	12) 1.00E+00	13) 1.00E+00	14) 1.00E+00	15) 1.00E+00

NORMALIZED AVERAGE WEIGHT

1) 1.00E+00	2) 1.00E+00	3) 1.00E+00	4) 1.00E+00	5) 0.00E+00
6) 1.00E+00	7) 1.00E+00	8) 1.00E+00	9) 1.00E+00	10) 1.00E+00
11) 1.00E+00	12) 1.00E+00	13) 1.00E+00	14) 1.00E+00	15) 1.00E+00

```
****************************************
*   C O M M E N T S      B L O C K    *
****************************************
```

1193.00 SECONDS EXECUTION TIME.

APPENDIX F

SINGLE COMPONENT RELIABILITY AND AVAILABILITY SIMULATION

RAMIN - EXPONENTIAL FAILURE DENSITY & EXPONENTIAL REPAIR DENSITY

```
NTYPE...1NTIME...5000N... 1K[illegible]1IPROB...1ISZE...10FI...  100.0   END
ICODE...0    LRN NO.1
KFDN... 1FPTR(1)...  .1000000E+03FPTR(2)...  .0000000E+00END
KRDN... 1RPTR(1)...  .1000000E+02RPTR(2)...  .0000000E+00END
```

SUBROUTINE SYST

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ***************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                              *
C *                                                             *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE              *
C * KST(I) - COMPONENT STATUS, 0 OR 1                           *
C ***************************************************************
C
      ISYSUP = KST(1)
C
         IF (ISYSUP.EQ.0) THEN
           J=J+1
           TEMP2(J) = TEMP1
         ENDIF
C
20    CONTINUE
C
C *********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C *********************************************
C
```

RAMOUT - EXPONENTIAL FAILURE DENSITY & EXPONENTIAL REPAIR DENSITY

RUN NO. 1

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)...............................         1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......        5000
NUMBER OF COMPONENTS IN THE SYSTEM.................          1
REPAIR SPECIFICATION (1,2 OR 3)....................          1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....         25
CLASS INTERVAL WIDTH...............................         10.00

COMP NO. 1 ICODE.. 0
          KFDN.. 1  FPTR(1).. .1000000E+03  FPTR(2).. .0000000E+00
          KRDN.. 1  RPTR(1).. .1000000E+02  RPTR(2).. .0000000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     98.70     SIGMA......     98.39
AVERAGE DNTIME...      9.78     SIGMA......      9.62

AVERAGE AVAILABILITY..  .91     SIGMA......       .00

THE 90 PERCENT CONFIDENCE LEVEL...   .91
THE 95 PERCENT CONFIDENCE LEVEL...   .91
```

RELIABILITY BLOCK

```
MEAN LIFE................     98.70
SIGMA....................     98.39
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	449	.904	.910
20.00	466	.808	.817
30.00	369	.733	.743
40.00	359	.660	.671
50.00	332	.594	.605
60.00	314	.531	.542
70.00	259	.479	.490
80.00	240	.431	.442
90.00	209	.389	.401
100.00	190	.351	.363
110.00	158	.320	.331
120.00	168	.287	.297
130.00	163	.255	.265
140.00	129	.229	.239
150.00	115	.206	.216
160.00	106	.185	.194
170.00	90	.168	.176
180.00	75	.153	.161
190.00	88	.136	.144
200.00	60	.124	.132
210.00	59	.112	.120
220.00	51	.103	.110
230.00	54	.092	.099
240.00	49	.083	.089
250.00	49	.073	.079
OVERFLOW	397		.000

NUMBER OF CLASS INTERVALS.....   25

```
CLASS INTERVAL WIDTH..........     10.00
MAXIMUM SYSTEM FAILURE TIME...    781.65
MINIMUM SYSTEM FAILURE TIME...       .02

FAILURE BLOCK

COMPONENT NO.     NUMBER OF FAILURES

     1                  5000
```

RAMOUT – NORMAL FAILURE DENSITY & EXPONENTIAL REPAIR DENSITY

RUN NO. 2

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)..................................      1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......   5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      1
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....     20
CLASS INTERVAL WIDTH...............................     10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 2  FPTR(1)..  .1000000E+03  FPTR(2)..  .3000000E+02
           KRDN.. 1  RPTR(1)..  .1000000E+02  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     98.90     SIGMA......     29.68
AVERAGE DNTIME...     10.02     SIGMA......      9.70

AVERAGE AVAILABILITY..  .91     SIGMA......       .00

THE 90 PERCENT CONFIDENCE LEVEL...   .91
THE 95 PERCENT CONFIDENCE LEVEL...   .91
```

RELIABILITY BLOCK

```
MEAN LIFE.................     98.90
SIGMA.....................     29.68
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	5	.998	.999
20.00	8	.996	.997
30.00	19	.992	.994
40.00	85	.973	.977
50.00	127	.946	.951
60.00	258	.893	.900
70.00	347	.821	.830
80.00	458	.728	.739
90.00	631	.601	.612
100.00	641	.473	.484
110.00	644	.344	.355
120.00	577	.230	.240
130.00	452	.141	.150
140.00	327	.078	.084
150.00	206	.038	.043
160.00	113	.017	.020
170.00	58	.007	.009
180.00	27	.002	.003
190.00	14	.000	.001
200.00	2	.000	.000
OVERFLOW	1		.000

```
NUMBER OF CLASS INTERVALS.....     20
CLASS INTERVAL WIDTH..........     10.00
MAXIMUM SYSTEM FAILURE TIME...    210.16
MINIMUM SYSTEM FAILURE TIME...       .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	5000

RAMOUT - WEIBULL FAILURE DENSITY (DECREASING FAILURE RATE) & EXPONENTIAL REPAIR DENSITY

RUN NO. 3

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)...........................      1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......  5000
NUMBER OF COMPONENTS IN THE SYSTEM.................     1
REPAIR SPECIFICATION (1,2 OR 3)....................     1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....    25
CLASS INTERVAL WIDTH...............................  10.00

COMP NO. 1 ICODE.. 0
          KFDN.. 3  FPTR(1)..  .1000000E+03  FPTR(2)..  .5000000E+00
          KRDN.. 1  RPTR(1)..  .1000000E+02  RPTR(2)..  .0000000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     194.21     SIGMA......     424.12
AVERAGE DNTIME...       9.78     SIGMA......       9.62

AVERAGE AVAILABILITY..  .95      SIGMA......        .01

THE 90 PERCENT CONFIDENCE LEVEL...   .95
THE 95 PERCENT CONFIDENCE LEVEL...   .95
```

RELIABILITY BLOCK

```
MEAN LIFE................     194.21
SIGMA....................     424.12
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	1355	.719	.729
20.00	458	.626	.637
30.00	324	.561	.573
40.00	223	.516	.528
50.00	210	.474	.486
60.00	162	.442	.454
70.00	137	.415	.426
80.00	122	.390	.402
90.00	98	.371	.382
100.00	98	.351	.363
110.00	92	.333	.344
120.00	55	.322	.333
130.00	67	.309	.320
140.00	78	.293	.304
150.00	79	.278	.288
160.00	72	.264	.274
170.00	52	.253	.264
180.00	52	.243	.253
190.00	39	.235	.245
200.00	58	.224	.234
210.00	34	.217	.227
220.00	30	.211	.221
230.00	40	.203	.213
240.00	40	.196	.205
250.00	32	.189	.199
OVERFLOW	993		.000

```
NUMBER OF CLASS INTERVALS.....     25
CLASS INTERVAL WIDTH..........  10.00
MAXIMUM SYSTEM FAILURE TIME...  6109.70
MINIMUM SYSTEM FAILURE TIME...     .00
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	5000

RAMOUT - WEIBULL FAILURE DENSITY (INCREASING FAILURE RATE) & EXPONENTIAL REPAIR DENSITY

```
                         RUN NO.  4

                         INPUT BLOCK

TYPE OF RUN (0,1,OR 2).............................      1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......   5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      1
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....     25
CLASS INTERVAL WIDTH...............................     10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1000000E+03  FPTR(2)..  .2000000E+01
           KRDN.. 1  RPTR(1)..  .1000000E+02  RPTR(2)..  .0000000E+00


                         OUTPUT BLOCK


AVAILABILITY BLOCK

AVERAGE UPTIME...     88.17     SIGMA......     45.78
AVERAGE DNTIME...      9.78     SIGMA......      9.62

AVERAGE AVAILABILITY..  .90     SIGMA......       .00

THE 90 PERCENT CONFIDENCE LEVEL...   .90
THE 95 PERCENT CONFIDENCE LEVEL...   .90

RELIABILITY BLOCK

MEAN LIFE................      88.17
SIGMA....................      45.78

RELIABILITY DISTRIBUTION
```

TIME	FREQUENCY	R95L	RMLE
10.00	46	.989	.991
20.00	139	.959	.963
30.00	219	.913	.919
40.00	324	.846	.854
50.00	383	.768	.778
60.00	399	.687	.698
70.00	438	.599	.610
80.00	430	.513	.524
90.00	430	.427	.438
100.00	379	.351	.363
110.00	344	.283	.294
120.00	327	.219	.228
130.00	248	.170	.179
140.00	212	.128	.136
150.00	156	.098	.105
160.00	161	.067	.073
170.00	106	.047	.052
180.00	72	.033	.037
190.00	58	.022	.026
200.00	41	.015	.018
210.00	29	.009	.012
220.00	19	.006	.008
230.00	13	.004	.005
240.00	12	.002	.003
250.00	6	.001	.002
OVERFLOW	9		.000

NUMBER OF CLASS INTERVALS..... 25
CLASS INTERVAL WIDTH.......... 10.00
MAXIMUM SYSTEM FAILURE TIME... 279.58
MINIMUM SYSTEM FAILURE TIME... 1.36

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	5000

APPENDIX G

PRODUCTION LINE RELIABILITY/AVAILABILITY SIMULATION

RAMIN - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES

```
NTYPE...1NTIME...5000N...19KFIX... 1IPROB... 1ISZE...15FI...   10.0   END
ICODE...0    LRU NO.1
KFDN... 3FPTR(1)...  .1140630E+04FPTR(2)...  .9000000E+00END
KRDN... 2RPTR(1)...  .2920000E+01RPTR(2)...  .5000000E+00END
ICODE...0    LRU NO.2
KFDN... 1FPTR(1)...  .1520960E+04FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2920000E+01RPTR(2)...  .5000000E+00END
ICODE...0    LRU NO.3
KFDN... 1FPTR(1)...  .1522100E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.4
KFDN... 1FPTR(1)...  .1522100E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.5
KFDN... 1FPTR(1)...  .1522100E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.6
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.7
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.8
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.9
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.10
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.11
KFDN... 1FPTR(1)...  .4555000E+02FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .2409000E+02RPTR(2)...  .4000000E+01END
ICODE...0    LRU NO.12
KFDN... 3FPTR(1)...  .1014700E+03FPTR(2)...  .1100000E+01END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.13
KFDN... 3FPTR(1)...  .1014700E+03FPTR(2)...  .1100000E+01END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.14
KFDN... 3FPTR(1)...  .1014700E+03FPTR(2)...  .1100000E+01END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.15
KFDN... 1FPTR(1)...  .2025800E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.16
KFDN... 1FPTR(1)...  .2025800E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.17
KFDN... 1FPTR(1)...  .2025800E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .1205000E+02RPTR(2)...  .2000000E+01END
ICODE...0    LRU NO.18
KFDN... 1FPTR(1)...  .1825000E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .4020000E+01RPTR(2)...  .6700000E+00END
ICODE...0    LRU NO.19
KFDN... 1FPTR(1)...  .1825000E+03FPTR(2)...  .0000000E+00END
KRDN... 2RPTR(1)...  .4020000E+01RPTR(2)...  .6700000E+00END
```

SUBROUTINE SYST

(ONLY THE RELEVANT SECTION OF THE PROGRAM IS SHOWN)

```
C
C ***************************************************************
C * ENTER THE SYSTEM FUNCTION HERE                              *
C *                                                             *
C * ISYSUP - SYSTEM STATUS, MAY BE LARGER THAN ONE              *
C * KST(I) - COMPONENT STATUS, 0 OR 1                           *
C ***************************************************************
C
      I1 = KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(3)*KST(6)*KST(12)*KST(17)*KST(19)
C
      I2 = KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(3)*KST(7)*KST(12)*KST(17)*KST(19)
C
      I3 = KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(4)*KST(8)*KST(13)*KST(17)*KST(19)
C
      I4 = KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(4)*KST(9)*KST(13)*KST(17)*KST(19)
C
      I5 = KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(5)*KST(10)*KST(14)*KST(17)*KST(19)
C
      I6 = KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(15)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(16)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(17)*KST(18)+
     &KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(15)*KST(19)+
     &KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(16)*KST(19)+
     &KST(1)*KST(2)*KST(5)*KST(11)*KST(14)*KST(17)*KST(19)
C
      ISYSUP=I1+I2+I3+I4+I5+I6
C
          IF (ISYSUP.EQ.0) THEN
            J=J+1
            TEMP2(J) = TEMP1
          ENDIF
C
20      CONTINUE
C
C ***********************************************
C * FIND SMALLEST SYSTEM TIME TO FAILURE SYSF *
C ***********************************************
C
```

RAMOUT - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES (10 HISTORIES)

```
                         RUN NO.  1

                         INPUT BLOCK

TYPE OF RUN (0,1,OR 2)...............................     1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     10
NUMBER OF COMPONENTS IN THE SYSTEM.................     19
REPAIR SPECIFICATION (1,2 OR 3)....................      1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....     15
CLASS INTERVAL WIDTH...............................     10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .[illegible]E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .[illegible]E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

AVERAGE UPTIME...	41.50	SIGMA......	15.69
AVERAGE DNTIME...	145.09	SIGMA......	40.16
AVERAGE AVAILABILITY..	.22	SIGMA......	.04

THE 90 PERCENT CONFIDENCE LEVEL... .19
THE 95 PERCENT CONFIDENCE LEVEL... .18

RELIABILITY BLOCK

MEAN LIFE.................. 41.50
SIGMA...................... 15.69

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	0	.741	1.000
20.00	1	.700	.900
30.00	1	.370	.800
40.0[illegible]	4	.082	.400
5[illegible].[illegible]	1	.035	.300
60.[illegible]0	2	.000	.100
70.00	0	.000	.100
80.00	1	.000	.000
90.00	0	.000	.000
100.00	0	.000	.000
110.00	0	.000	.000
120.00	0	.000	.000
130.00	0	.000	.000
140.00	0	.000	.000
150.00	0	.000	.000
OVERFLOW	0		.000

NUMBER OF CLASS INTERVALS..... 15
CLASS INTERVAL WIDTH.......... 10.00
MAXIMUM SYSTEM FAILURE TIME... 71.90
MINIMUM SYSTEM FAILURE TIME... 19.74

FAILURE BLOCK

COMPONENT NO. NUMBER OF FAILURES

1	0
2	0
3	3
4	3
5	1
6	5
7	6
8	9
9	8
10	8
11	8
12	6
13	2
14	4
15	2
16	3
17	3
18	4
19	6

RAMOUT - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES (50 HISTORIES)

RUN NO. 1

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)...........................         1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......      50
NUMBER OF COMPONENTS IN THE SYSTEM.................      19
REPAIR SPECIFICATION (1,2 OR 3)....................       1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....      15
CLASS INTERVAL WIDTH...............................   10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

                       OUTPUT BLOCK

AVAILABILITY BLOCK

AVERAGE [illegible]TIME...      50.40     SIGMA......      26.99
AVERAGE [illegible]TIME...     133.75     SIGMA......      38.54

AVERAGE AVAILABILITY..  .27     SIGMA......        .02

THE 90 PERCENT CONFIDENCE LEVEL...   .25
THE 95 PERCENT CONFIDENCE LEVEL...   .25

RELIABILITY BLOCK

MEAN LIFE................      50.40
SIGMA....................      26.99

RELIABILITY DISTRIBUTION

     TIME     FREQUENCY    R95L      RMLE

     10.00        3        .874      .940
     20.00        2        .830      .900
     30.00        4        .731      .820
     40.00        8        .550      .660
     50.00       11        .325      .440
     60.00        9        .158      .260
     70.00        5        .075      .160
     80.00        2        .039      .120
     90.00        4        .001      .040
    100.00        0        .001      .040
    110.00        0        .001      .040
    120.00        0        .001      .040
    130.00        1        .000      .020
    140.00        0        .000      .020
    150.00        0        .000      .020

OVERFLOW       1                     .000

NUMBER OF CLASS INTERVALS......      15
CLASS INTERVAL WIDTH...........   10.00
MAXIMUM SYSTEM FAILURE TIME...   154.62
MINIMUM SYSTEM FAILURE TIME...     8.49

FAILURE BLOCK

COMPONENT NO.     NUMBER OF FAILURES
```

1	1
2	1
3	12
4	16
5	9
6	29
7	26
8	40
9	34
10	37
11	35
12	26
13	18
14	18
15	12
16	11
17	13
18	12
19	18

RAMOUT - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES (100 HISTORIES)

RUN NO. 1

INPUT BLOCK

```
TYPE OF RUN (0,1,OR 2)............................       1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     100
NUMBER OF COMPONENTS IN THE SYSTEM.................      19
REPAIR SPECIFICATION (1,2 OR 3)....................       1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....      15
CLASS INTERVAL WIDTH...............................   10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...      50.43     SIGMA......      24.99
AVERAGE DNTIME...     134.36     SIGMA......      42.38

AVERAGE AVAILABILITY..  .27      SIGMA......        .01

THE 90 PERCENT CONFIDENCE LEVEL...   .26
THE 95 PERCENT CONFIDENCE LEVEL...   .25
```

RELIABILITY BLOCK

```
MEAN LIFE..................     50.43
SIGMA......................     24.99
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	4	.923	.960
20.00	6	.851	.900
30.00	9	.745	.810
40.00	14	.593	.670
50.00	21	.378	.460
60.00	17	.215	.290
70.00	14	.091	.150
80.00	3	.067	.120
90.00	8	.014	.040
100.00	0	.014	.040
110.00	1	.008	.030
120.00	1	.004	.020
130.00	1	.001	.010
140.00	0	.001	.010
150.00	0	.001	.010
OVERFLOW	1		.000

```
NUMBER OF CLASS INTERVALS.....      15
CLASS INTERVAL WIDTH..........   10.00
MAXIMUM SYSTEM FAILURE TIME...  154.62
MINIMUM SYSTEM FAILURE TIME...    4.62
```

FAILURE BLOCK

COMPONENT NO.   NUMBER OF FAILURES

1	4
2	2
3	30
4	32
5	20
6	64
7	56
8	74
9	69
10	69
11	65
12	51
13	42
14	45
15	19
16	22
17	31
18	24
19	32

RAMOUT - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES (5000 HISTORIES)

```
                              RUN NO.  1

                              INPUT BLOCK

TYPE OF RUN (0,1,OR 2)..............................        1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM.................       19
REPAIR SPECIFICATION (1,2 OR 3)....................        1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       15
CLASS INTERVAL WIDTH...............................     10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     53.67    SIGMA......     28.87
AVERAGE DNTIME...    135.40    SIGMA......     39.97

AVERAGE AVAILABILITY..  .28    SIGMA......      .00

THE 90 PERCENT CONFIDENCE LEVEL...    .28
THE 95 PERCENT CONFIDENCE LEVEL...    .28
```

RELIABILITY BLOCK

```
MEAN LIFE.................       53.67
SIGMA.....................       28.87
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	109	.975	.978
20.00	328	.906	.913
30.00	571	.789	.798
40.00	774	.632	.644
50.00	811	.470	.481
60.00	649	.340	.352
70.00	555	.231	.241
80.00	393	.153	.162
90.00	296	.096	.103
100.00	173	.062	.068
110.00	115	.040	.045
120.00	83	.025	.029
130.00	48	.016	.019
140.00	41	.008	.011
150.00	21	.005	.007
OVERFLOW	33		.000

```
NUMBER OF CLASS INTERVALS.....      15
CLASS INTERVAL WIDTH..........     10.00
MAXIMUM SYSTEM FAILURE TIME...    217.58
MINIMUM SYSTEM FAILURE TIME...       .03
```

FAILURE BLOCK

COMPONENT NO.   NUMBER OF FAILURES

1	199
2	180
3	1444
4	1532
5	1533
6	3453
7	3466
8	3377
9	3384
10	3384
11	3309
12	2197
13	2169
14	2142
15	1168
16	1115
17	1220
18	1312
19	1322

RAMOUT - COMPONENTS SWITCHED OFF DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE MAXIMUM COMPONENT REPAIR TIME

```
                         RUN NO.  2

                        INPUT BLOCK

TYPE OF RUN (0,1,OR 2)..............................       1
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM..................      19
REPAIR SPECIFICATION (1,2 OR 3).....................       2
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       15
CLASS INTERVAL WIDTH................................   10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .15[illegible]100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     53.67     SIGMA......     28.87
AVERAGE DNTIME...     27.68     SIGMA......      4.01

AVERAGE AVAILABILITY..  .66     SIGMA......       .00

THE 90 PERCENT CONFIDENCE LEVEL...     .66
THE 95 PERCENT CONFIDENCE LEVEL...     .66
```

RELIABILITY BLOCK

```
MEAN LIFE..................     53.67
SIGMA......................     28.87
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	109	.975	.978
20.00	328	.906	.913
30.00	571	.789	.798
40.00	774	.632	.644
50.00	811	.470	.481
60.00	649	.340	.352
70.00	555	.231	.241
80.00	393	.153	.162
90.00	296	.096	.103
100.00	173	.062	.068
110.00	[illegible]	.040	.045
120.00	[illegible]	.025	.029
130.00	48	.016	.019
140.00	41	.008	.011
150.00	21	.005	.007
OVERFLOW	33		.000

```
NUMBER OF CLASS INTERVALS.....       15
CLASS INTERVAL WIDTH..........    10.00
MAXIMUM SYSTEM FAILURE TIME...   217.58
MINIMUM SYSTEM FAILURE TIME...      .03
```

FAILURE BLOCK

COMPONENT NO.   NUMBER OF FAILURES

1	199
2	180
3	1444
4	1552
5	1533
6	3453
7	3466
8	3377
9	3384
10	3384
11	3309
12	2197
13	2169
14	2142
15	1168
16	1115
17	1220
18	1312
19	1322

RAMOUT - COMPONENTS LEFT ON DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE SUM OF COMPONENT REPAIR TIMES

```
                              RUN NO.  3

                              INPUT BLOCK

TYPE OF RUN (0,1,OR 2).............................          2
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......       5000
NUMBER OF COMPONENTS IN THE SYSTEM.................         19
REPAIR SPECIFICATION (1,2 OR 3)....................          1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....         15
CLASS INTERVAL WIDTH...............................      10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     13.86     SIGMA......     22.52
AVERAGE DNTIME...    123.92     SIGMA......     38.43

AVERAGE AVAILABILITY..  .10     SIGMA......       .01

THE 90 PERCENT CONFIDENCE LEVEL...    .10
THE 95 PERCENT CONFIDENCE LEVEL...    .10
```

RELIABILITY BLOCK

```
MEAN LIFE................      13.86
SIGMA....................      22.52
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	3184	.352	.363
20.00	509	.251	.261
30.00	410	.170	.179
40.00	280	.116	.123
50.00	220	.073	.079
60.00	136	.047	.052
70.00	90	.030	.034
80.00	69	.017	.020
90.00	41	.010	.012
100.00	20	.006	.008
110.00	16	.003	.005
120.00	8	.002	.003
130.00	5	.001	.002
140.00	2	.001	.002
150.00	1	.001	.002
OVERFLOW	9		.000

```
NUMBER OF CLASS INTERVALS.....      15
CLASS INTERVAL WIDTH.........    10.00
MAXIMUM SYSTEM FAILURE TIME...  204.74
MINIMUM SYSTEM FAILURE TIME...     .00
```

FAILURE BLOCK

COMPONENT NO.   NUMBER OF FAILURES

1	512
2	438
3	1991
4	1992
5	1968
6	2657
7	2666
8	2655
9	2656
10	2675
11	2673
12	2357
13	2325
14	2336
15	1666
16	1712
17	1692
18	1868
19	1828

RAMOUT - COMPONENTS LEFT ON DURING SYSTEM REPAIR WITH THE SYSTEM REPAIR TIME EQUAL TO THE MAXIMUM COMPONENT REPAIR TIME

```
                      RUN NO.  4

                      INPUT BLOCK

TYPE OF RUN (0,1,OR 2)..............................      2
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......    5000
NUMBER OF COMPONENTS IN THE SYSTEM.................      19
REPAIR SPECIFICATION (1,2 OR 3)....................       2
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....      15
CLASS INTERVAL WIDTH...............................   10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01
```

```
COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

AVAILABILITY BLOCK

```
AVERAGE UPTIME...     40.12     SIGMA......     30.07
AVERAGE DNTIME...     27.05     SIGMA......      4.71

AVERAGE AVAILABILITY..  .60     SIGMA......       .00

THE 90 PERCENT CONFIDENCE LEVEL...   .59
THE 95 PERCENT CONFIDENCE LEVEL...   .59
```

RELIABILITY BLOCK

```
MEAN LIFE................    40.12
SIGMA....................    30.07
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	759	.840	.848
20.00	614	.715	.725
30.00	740	.566	.577
40.00	693	.427	.439
50.00	633	.301	.312
60.00	475	.208	.217
70.00	339	.141	.149
80.00	245	.093	.100
90.00	163	.062	.068
100.00	112	.041	.045
110.00	78	.026	.030
120.00	63	.014	.017
130.00	35	.008	.010
140.00	23	.004	.006
150.00	9	.002	.004
OVERFLOW	19		.000

```
NUMBER OF CLASS INTERVALS.....      15
CLASS INTERVAL WIDTH..........   10.00
MAXIMUM SYSTEM FAILURE TIME...  259.12
MINIMUM SYSTEM FAILURE TIME...     .00
```

FAILURE BLOCK

COMPONENT NO.   NUMBER OF FAILURES

1	297
2	232
3	1599
4	1621
5	1618
6	3118
7	3133
8	3126
9	3139
10	3087
11	3151
12	2169
13	2195
14	2200
15	1302
16	1224
17	1249
18	1385
19	1382

RAMOUT - PURE RELIABILITY SIMULATION

```
                         RUN NO.  5

                         INPUT BLOCK

TYPE OF RUN (0,1,OR 2)...............................      0
REQUIRED NUMBER OF SIMULATED SYSTEM FAILURES.......     5000
NUMBER OF COMPONENTS IN THE SYSTEM.................       19
REPAIR SPECIFICATION (1,2 OR 3)....................        1
NUMBER OF CLASS INTERVALS FOR RELIABILITY CALC.....       15
CLASS INTERVAL WIDTH...............................    10.00

COMP NO. 1 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1140630E+04  FPTR(2)..  .9000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 2 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1520960E+04  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2920000E+01  RPTR(2)..  .5000000E+00

COMP NO. 3 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR[illegible]  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 4 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 5 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .1522100E+03  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO. 6 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPT[illegible]  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPT[illegible]  .4000000E+01

COMP NO. 7 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 8 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO. 9 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.10 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.11 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .4555000E+02  FPTR(2)..  .0000000E+00
           KRDN.. 2  RPTR(1)..  .2409000E+02  RPTR(2)..  .4000000E+01

COMP NO.12 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.13 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.14 ICODE.. 0
           KFDN.. 3  FPTR(1)..  .1014700E+03  FPTR(2)..  .1100000E+01
           KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.15 ICODE.. 0
           KFDN.. 1  FPTR(1)..  .2025[illegible]  FPTR(2)..  .0000000E+00
```

```
          KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.16 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
          KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.17 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .2025800E+03  FPTR(2)..  .0000000E+00
          KRDN.. 2  RPTR(1)..  .1205000E+02  RPTR(2)..  .2000000E+01

COMP NO.18 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
          KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00

COMP NO.19 ICODE.. 0
          KFDN.. 1  FPTR(1)..  .1825000E+03  FPTR(2)..  .0000000E+00
          KRDN.. 2  RPTR(1)..  .4020000E+01  RPTR(2)..  .6700000E+00
```

OUTPUT BLOCK

RELIABILITY BLOCK

```
MEAN LIFE................      54.38
SIGMA....................      29.74
```

RELIABILITY DISTRIBUTION

TIME	FREQUENCY	R95L	RMLE
10.00	139	.968	.972
20.00	322	.901	.908
30.00	530	.793	.802
40.00	738	.643	.654
50.00	796	.483	.495
60.00	690	.346	.357
70.00	524	.242	.252
80.00	398	.164	.173
90.00	285	.108	.116
100.00	200	.069	.076
110.00	128	.045	.050
120.00	91	.028	.032
130.00	56	.017	.021
140.00	29	.012	.015
150.00	25	.008	.010
OVERFLOW	49		.000

```
NUMBER OF CLASS INTERVALS.....    15
CLASS INTERVAL WIDTH..........    10.00
MAXIMUM SYSTEM FAILURE TIME...   220.40
MINIMUM SYSTEM FAILURE TIME...      .08
```

FAILURE BLOCK

COMPONENT NO.	NUMBER OF FAILURES
1	336
2	177
3	1492
4	1535
5	1496
6	3404
7	3396
8	3435
9	3362
10	3412
11	3437
12	1980

13	2080
14	1995
15	1167
16	1125
17	1168
18	1294
19	1288

APPENDIX H

SPAR® SIMULATION RESULTS – PRODUCTION LINE

STIN – PRODUCTION LINE

```
RECORD-1.1 TITLE
PRODUCTION LINE EXAMPLE
DEFAULT - COMPONENTS ACTIVE DURING SYSTEM REPAIR
DEFAULT - SYSTEM REPAIR TIME IS EQUAL TO MAX COMPONENT REPAIR TIME
SYSTEM CHECKUP LEVEL - COMPONENTS REPAIRED FOLLOWING SYSTEM FAILURE
CONTINUOUS CHECKUP MODE - SYSTEM CHECKED AT EACH STOCHASTIC EVENT
RECORD-1.2  MODE OF RUN
2

RECORD-1.3   NPS
500

RECORD-1.4  FLAG OF NON-EXPONENTIAL FIELD DISTRIBUTIONS
1

RECORD-2.1    SERVICE TIME
10000

RECORD-2.2  NUMBER OF DIFFERENT PROFILE STATES
1

RECORD-2.3  TIME POINTS OF MISSION PROFILE FLIPS

RECORD-2.4  PROFILE STATES IN PROFILE DEFINITION

RECORD-2.5  NUMBER OF SYSTEMS AT T=0
1

RECORD-2.6 NUMBER OF TIME POINTS FOR SYSTEMS ACQUISITION

RECORD-2.7 TIME POINTS OF SYSTEMS ACQUISITION

RECORD-2.8  NUMBER OF SYSTEMS ADDED AT EACH TIME POINT

RECORD-3.1  SYSTEM RELIABILITY MODEL
2

RECORD-3.2  NUMBER OF LRU'S IN SYSTEM
19

RECORD-3.3  NUMBER OF DIFFERENT LRU TYPES
7

RECORD-3.4  LRU TYPE IDENTIFICATION
1         A
2         B
3         C
4         D
5         E
6         F
7         G

RECORD-3.5  SYSTEM COMPOSITION OF LRU'S
1  2  3  3  3  4  4  4  4   4   4   5   5   5   6   6  6   7   7

RECORD-4.1  LRU TYPE MEAN REPLACEMENT TIME (AT LEVEL A)

RECORD-4.2  LRU A TO B SHIPMENT TIME DISTR.  (C,E,N)
C  C  C  C  C  C  C

RECORD-4.3  LRU A TO B SHIPMENT TIME
1.E-06
```

```
1.E-06
1.E-06
1.E-06
1.E-06
1.E-06
1.E-06

RECORD-4.4  LRUS PASSIVE FAILURE RATES FLAG

RECORD-4.5  FAILURE RATES

RECORD-4.6  LRU REPAIR (AT LEVEL B) TIME DISTRIBUTION
1   C      1.E-06
2   C      1.E-06
3   C      1.E-06
4   C      1.E-06
5   C      1.E-06
6   C      1.E-06
7   C      1.E-06

RECORD-4.7 TYPE NUMBERS OF LRUS REPAIRED AT LEVEL A
1  2  3  4  5  6  7

RECORD-4.8  MAXIMUM NUMBER OF LRU REPAIR CYCLES

RECORD-4.9  PROBABILITY TO FIND A FAILED SPARE AT LEVEL A

RECORD-5.1  NUMBER OF TIME POINTS FOR LRU'S ACQUISITION

RECORD-5.2  TIME POINTS OF LRU'S ACQUISITION

RECORD-5.3  LRU STORAGE
1000
1000
1000
1000
1000
1000
1000

RECORD-5.4  FLAG OF LRU PRICE CONSIDERATIONS

RECORD-5.5  PRICES PER LRU TYPE AT ACQUISITION TIME

RECORD-5.6  CURRENCY

RECORD-6.1  CHECK-UP LEVEL AND MODE
S
C

RECORD-6.2  CHECK-UP CYCLES SPECIFICATION

RECORD-6.3  TEST COVERAGE VALUES FOR EACH LRU TYPE

RECORD-6.4  TEST EFFICIENCY VALUES FOR EACH LRU TYPE

RECORD-7.1  NUMBER OF TIME POINTS FOR AVAILABILITY CALCULATION
1

RECORD-7.2  TIME POINTS FOR AVAILABILITY CALCULATION
10000

RECORD-7.3  RISK FUNCTION

RECORD-7.4  PROBABILITY OF K SYSTEMS UP - VALUES OF K ARE :

RECORD-8.1  REPLACEMENT TIME DISTRIBUTIONS
PS  1
1   N   2.92  0.50
2   N   2.92  0.50
3   N   12.05 2.00
```

4	N	24.09	4.00
5	N	12.05	2.00
6	N	12.05	2.00
7	N	4.02	0.67

RECORD-8.2 FAILURE TIME DISTRIBUTIONS OF LRUS IN ACTIVE STATE

PS 1

1	W	1.772E-03	0.9
2	W	657.5E-06	1
3	W	6.57E-03	1
4	W	21.95E-03	1
5	W	6.209E-03	1.1
6	W	4.936E-03	1
7	W	5.479E-03	1

RECORD-8.3 FAILURE TIME DISTRIBUTIONS OF LRUS IN PASSIVE STATE

LBOUT - PRODUCTION LINE

```
      FUNCTION NSYSCM()
      NSYSCM =  19
      RETURN
      END
      FUNCTION ISYSUP(J,K)
      DIMENSION N(19),K(1)
      ISYSUP=0
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+15)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+15)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+16)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+16)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+16)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+16)
     1 *K(J+19)
```

```
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+16)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+16)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+16)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+17)
     1 *K(J+18)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+6)*K(J+12)*K(J+17)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+3)*K(J+7)*K(J+12)*K(J+17)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+8)*K(J+13)*K(J+17)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+4)*K(J+9)*K(J+13)*K(J+17)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+10)*K(J+14)*K(J+17)
     1 *K(J+19)
      IF(ISYSUP.GT.0.)RETURN
      ISYSUP= K(J+1)*K(J+2)*K(J+5)*K(J+11)*K(J+14)*K(J+17)
     1 *K(J+19)
      RETURN
      END
```

STOUT - PRODUCTION LINE

```
PRODUCTION LINE EXAMPLE
DEFAULT - COMPONENTS ACTIVE DURING SYSTEM REPAIR
DEFAULT - SYSTEM REPAIR TIME IS EQUAL TO MAX COMPONENT REPAIR TIME
SYSTEM CHECKUP LEVEL - COMPONENTS REPAIRED FOLLOWING SYSTEM FAILURE
CONTINUOUS CHECKUP MODE - SYSTEM CHECKED AT EACH STOCHASTIC EVENT


        ****************************
        * I N P U T   B L O C K *
        ****************************


     G E N E R A L   C O N T R O L   R E C O R D S
     ----------------------------------------------

MODE OF RUN (1-STORAGE REQUIREMENTS; 2-DEFINED STORAGE)............ 2
NUMBER OF HISTORIES ............................................  500
NUMBER OF HISTORIES USED TO TALLY DISTRIBUTIONS ................  500
NON-EXPONENTIAL FIELD DISTRIBUTIONS ARE INCLUDED ..................


     F I E L D   D E S C R I P T I O N   R E C O R D S
     --------------------------------------------------

SERVICE TIME ............................................ 1.0000E+04
NUMBER OF DIFFERENT MISSION PROFILE STATES ....................... 1

NUMBER OF CHANGE PROFILE TIME-POINTS : ........................    0

THERE ARE NO CHANGE PROFILE TIME-POINTS

NUMBER OF STATES IN PROFILE DEFINITION : ......................    1

PROFILE STATES ARE NOT GIVEN IN INPUT,
   PROFILE STATE NO. 1  IS ASSUMED
FROM THE BEGINNING TO THE END OF LIFE


NUMBER OF SYSTEMS IN THE FIELD AT  T = 0.......................    1

NUMBER OF TIME POINTS FOR SYSTEMS ACQUISITION ..................  0


     S Y S T E M   D E S C R I P T I O N   R E C O R D S
     ----------------------------------------------------

SYSTEM RELIABILITY MODEL  ( 1 - SERIAL ; 2 - NETWORK).............. 2
NUMBER OF LRU"S IN SYSTEM.......................................   19
NUMBER OF DIFFERENT LRU TYPES .................................... 7

LRU TYPES IDENTIFICATION TABLE

LRU TYPE                 LRU TYPE
 NUMBER               IDENTIFICATION
                           NAME

     1               A
     2               B
     3               C
     4               D
     5               E
     6               F
     7               G


SYSTEM COMPOSITION OF LRU'S
   1    2    3    3    3    4    4    4    4    4    4    5    5    5
   6    6    6    7    7
```

L R U   D E S C R I P T I O N   R E C O R D S

LRU TYPE REPLACEMENT TIME DISTRIBUTION

PROFILE STATE NO. 1

LRU TYPE NUMBER	DISTRIBUTION TYPE	FIRST PARAMETER	SECOND PARAMETER
1	NORMAL	2.920E+00	5.000E-01
2	NORMAL	2.920E+00	5.000E-01
3	NORMAL	1.205E+01	2.000E+00
4	NORMAL	2.409E+01	4.000E+00
5	NORMAL	1.205E+01	2.000E+00
6	NORMAL	1.205E+01	2.000E+00
7	NORMAL	4.020E+00	6.700E-01

LRU A TO B SHIPMENT TIME DISTRIBUTION

LRU TYPE NUMBER	DISTRIBUTION TYPE	FIRST PARAMETER	SECOND PARAMETER
1	CONSTANT	1.000E-06	
2	CONSTANT	1.000E-06	
3	CONSTANT	1.000E-06	
4	CONSTANT	1.000E-06	
5	CONSTANT	1.000E-06	
6	CONSTANT	1.000E-06	
7	CONSTANT	1.000E-06	

LRU TYPE FAILURE TIME DISTRIBUTIONS IN ACTIVE STATE

PROFILE STATE NO. 1

LRU TYPE NUMBER	DISTRIBUTION TYPE	FIRST PARAMETER	SECOND PARAMETER
1	WEIBULL	1.772E-03	9.000E-01
2	WEIBULL	4.575E-04	1.000E+00
3	WEIBULL	6.570E-03	1.000E+00
4	WEIBULL	2.195E-02	1.000E+00
5	WEIBULL	6.209E-03	1.100E+00
6	WEIBULL	4.936E-03	1.000E+00
7	WEIBULL	5.479E-03	1.000E+00

LRU TYPE FAILURE TIME DISTRIBUTIONS IN PASSIVE STATE ARE NOT GIVEN

LRU REPAIR (AT LEVEL B) TIME DISTRIBUTION

LRU TYPE NUMBER	REPAIR TIME DISTRIBUTION TYPE	FIRST PARAMETER	SECOND PARAMETER
1	CONSTANT	1.000E-06	
2	CONSTANT	1.000E-06	
3	CONSTANT	1.000E-06	
4	CONSTANT	1.000E-06	
5	CONSTANT	1.000E-06	
6	CONSTANT	1.000E-06	
7	CONSTANT	1.000E-06	

TYPE NUMBERS OF LRU"S REPAIRED AT LEVEL A (NO REPLACEMENT)
1 2 3 4 5 6 7

MAXIMUM NUMBER OF LRU REPAIR CYCLES

LRU TYPE MAXIMUM

NUMBER	NUMBER OF CYCLES
1	UNLIMITED
2	UNLIMITED
3	UNLIMITED
4	UNLIMITED
5	UNLIMITED
6	UNLIMITED
7	UNLIMITED

PROBABILITY TO FIND FAILED SPARES AT LEVEL A IS NOT GIVEN,
IT IS ASSUMED EQUAL TO ZERO FOR ALL LRU TYPES.

S P A R E   P A R T S   S T R A T E G Y   R E C O R D S

NUMBER OF TIME POINTS FOR LRU"S ACQUISITION 0

LRU STORAGE STRATEGY
(FIRST ENTRY STANDS FOR NUMBER OF LRUS STORED AT TIME = 0)

LRU TYPE NUMBER	N U M B E R O F L R U S
1	1000
2	1000
3	1000
4	1000
5	1000
6	1000
7	1000

FLAG OF LRU"S PRICES CONSIDERATION 0

CURRENCY IS NOT SPECIFIED

M A I N T E N A N C E   P O L I C Y   R E C O R D S

C H E C K - U P   P O L I C Y :

SYSTEM LEVEL OF CHECK-UP
CONTINUOUS CHECK-UP

T A L L Y   R E C O R D S

NUMBER OF POINTS FOR AVAILABILITY CALCULATION 1

TIME POINTS FOR AVAILABILITY CALCULATION
1.000E+04

1

****
* A V A I L A B I L I T Y   B L O C K *
****

AVAILABILITY WITH DEFINED SPARE PARTS STORAGE

POINTS	AT THE POINT	P.R.S.D.	IN THE INTERVAL	P.R.S.D.
10000.00	6.1200000E-01	3.56%	6.1512023E-01	3.49%

1

```
*********************************************
*  SPARE  PARTS  CHARACTERISTICS   BLOCK  *
*********************************************
```

L R U TYPE NUMBER	PROBABILITY OF SPARE PART SHORTAGE IN LIFE HISTORY	AVAILABILITY OF THE SPARE PARTS	AVERAGE WAITING TIME
1	0.0000000E+00	1.0000000E+00	0.0000000E+00
2	0.0000000E+00	1.0000000E+00	0.0000000E+00
3	0.0000000E+00	1.0000000E+00	0.0000000E+00
4	0.0000000E+00	1.0000000E+00	0.0000000E+00
5	0.0000000E+00	1.0000000E+00	0.0000000E+00
6	0.0000000E+00	1.0000000E+00	0.0000000E+00
7	0.0000000E+00	1.0000000E+00	0.0000000E+00

1

```
*****************************************
*S E N S I T I V I T Y    B L O C K*
*****************************************
```

L R U   TYPE SENSITIVITY FOR DEFINED SPARE PARTS STORAGE

TOTAL NUMBER OF SYSTEMS FAILURES : 1205

LRU TYPE	FAILED PER HISTORY	UNAVAILABILITY SENSITIVITY	SYSTEMS FAILURES UPON EACH TYPE	FAILURE SENSITIVITY
1	0.16	1.03178E-02	66	5.47718E-02
2	0.11	1.53400E-02	42	3.48548E-02
3	2.49	1.65398E-01	220	1.82573E-01
4	9.75	4.94490E-01	466	3.86722E-01
5	3.24	2.62888E-01	283	2.34855E-01
6	1.85	1.04144E-02	27	2.24066E-02
7	1.41	4.11324E-02	101	8.38174E-02

NUMBER OF SYSTEMS DOWN-TIMES USED TO BUILD DISTRIBUTION : 500

%	SYSTEMS DOWN TIME
0.05	2.147949E+00
0.10	2.788532E+00
0.15	3.279778E+00
0.20	3.574600E+00
0.25	4.238621E+00
0.30	7.941780E+00
0.35	9.580261E+00
0.40	1.033606E+01
0.45	1.095369E+01
0.50	1.159746E+01
0.55	1.222325E+01
0.60	1.285759E+01
0.65	1.396498E+01
0.70	1.582735E+01
0.75	1.915582E+01
0.80	2.168119E+01
0.85	9.789642E+03
0.90	9.855004E+03
0.95	9.883146E+03

0.99 9.984993E+03

1

```
*******************************************
*  D I A G N O S T I C S     B L O C K  *
*******************************************
```

AVERAGE NO. OF COLLISIONS PER HISTORY	:	31.512
AVERAGE NO. OF SYSTEMS FAILURES PER HISTORY	:	2.410
NUMBER OF FORWARD SAMPLINGS		261482
NUMBER OF FORWARD SAMPLING REJECTIONS		0
EXECUTION TIME IN SECONDS	:	464.639

1

BIBLIOGRAPHY

1. Barry, T.M. *Electrical Power System – Preliminary Operational Reliability Analysis*, Boeing Commercial Aircraft Company – Coordination Sheet, no. R69–96, April 18, 1969.

2. Bernstein, N. *Reliability and Supportability Definitions and Parameters*, Israel Aircraft Industries Ltd, Training Plant, 1988, pp. 93–100.

3. Blanchard, B.S. and Fabrycky, W.J. *Systems Engineering and Analysis*, Prentice–Hall, 1981, pp. 279–280.

4. Billinton, R. and Allan, R.N. *Reliability Evaluation of Engineering Systems: Concepts and Techniques*, Longman Scientific and Technical, 1987, pp. 62–63.

5. Caplen, R. *A Practical Approach to Reliability*, Business Books Limited, 1972, pg. 5.

6. Caplen, R. *A Practical Approach to Reliability*, Business Books Limited, 1972, pp. 55–57.

7. Dhillon, B.S. *Reliability Engineering in Systems Design and Operation*, Van Nostrand Reinhold Company, 1983, pp. 291–293.

8. Dubi, Prof A. *Introduction to the RMC–1 Monte Carlo Multipurpose Code Package*, Malchi Science, Israel, pp. 39–44.

9. Dubi, Prof A. *Spar 1.2 N User's Guide*, Malchi Science, Israel, 1989, Chapter 2, pp. 4–6.

10. Evans, R. Mean Time Between Failure, *IEEE Transactions on Reliability*, Vol 35, No 3, August 1986, pg. 225.

11. Fielding, J.P. and Hussain, A.M. *The Use of Reliability Techniques in a Transport Aircraft Fuel System Design Study*, Proceedings of the 8th Advances in Reliability Technology Symposium, 1984.

12. Fielding, J.P. and Meng, X.Z. *The Use of Reliability Techniques in a Combat Aircraft Flying Control System Design Study*, Journal of Reliability Engineering, vol. 15, 1986, pp 61–78.

13. Forry, K.E. *A General Monte Carlo Simulation Model for Estimating Large Scale System Reliability and Availability*, PhD Thesis, Clemson University, 1972.

14. Goldfeld, Dr A. and Dubi, Prof A. *Monte Carlo Methods in the Computation of System Reliability*, Energia Nucleare, vol. 4, no. 3,

September–December 1987.

15. Hines, W.W and Montgomery, D.C. *Probability and Statistics in Engineering and Management Science*, 2nd ed. John Wiley and Sons, 1980, pp. 185–188

16. Hines, W.W and Montgomery, D.C. *Probability and Statistics in Engineering and Management Science*, 2nd ed. John Wiley and Sons, 1980, pp. 240–241

17. Lewis, P.A.W. and Orav, E.J. *Simulation Methodology for Statisticians, Operations Analysts and Engineers*, Wadsworth & Brooks/Cole, 1989, volume 1, pp. 89–93.

18. Nowlan, F.S and Heap, H.F *Reliability-Centered Maintenance*, United Airlines, pp. 34–39.

19. O'Connor, P.D. *Practical Reliability Engineering*, 2nd. ed. Wiley, 1989, pp. 39–40.

20. Smith, D.J. and Babb, A.H. *Maintainability Engineering*, Pitman Publishing, 1973, pp. 21–23.

21. Sparrius, A. *Life Cycle Product Support Course Notes*, Ad Sparrius System Engineering and Management (Pty) Ltd, South Africa, April 1991, chapter 1, pp. 8–11.

22. Taha, H.A. *Operations Research an Introduction*, 4th ed. Maxwell Macmillan International Editions, 1989, pp. 710–713.

23. Watras, H.J. *Combining a level of repair model with an availability centered provisioning model for logistic support analysis*, Msc Thesis, Naval Postgraduate School, Monterey, California, September 1983, pp. 27–30.