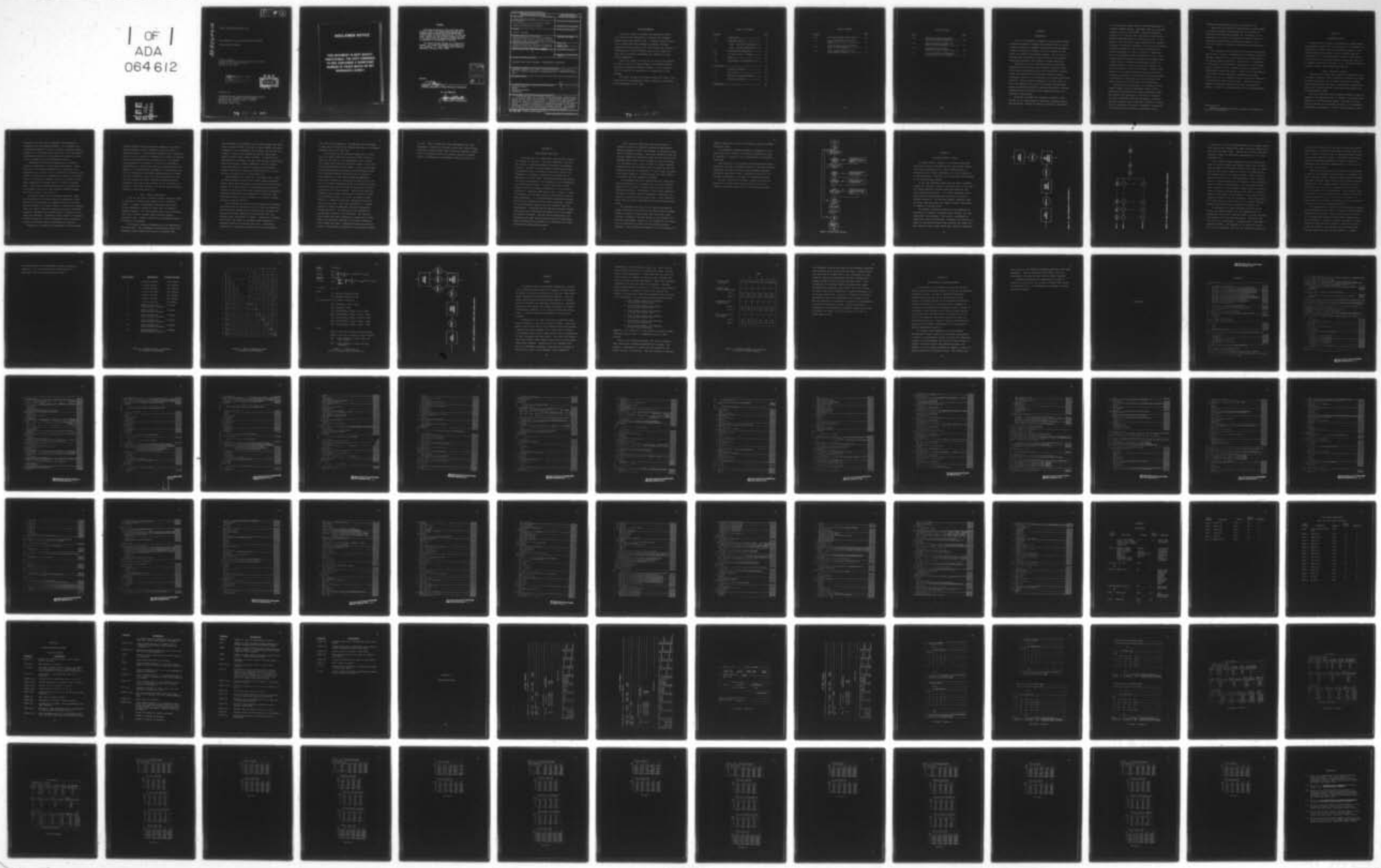
AD-A064 612   DARCOM INTERN TRAINING CENTER TEXARKANA TEX   F/G 13/8
THE SIMULATION AND ANALYSIS OF MULTI-PRODUCT MANUFACTURING SYST--ETC(U)
APR 76   A S HARTMAN
UNCLASSIFIED   DARCOM-ITC-02-08-76-215   NL

1 OF 1
ADA
064612



LEVEL II (1)

AD-A064612

Report DARCOM-ITC-02-08-76-215

THE SIMULATION AND ANALYSIS OF MULTI-PRODUCT MANUFACTURING SYSTEMS

Alan S. Hartman
Product/Production Graduate Engineering Program
DARCOM Intern Training Center
Red River Army Depot

Prepared for

PRODUCT/PRODUCTION GRADUATE ENGINEERING PROGRAM
AND TEXAS A&M UNIVERSITY GRADUATE CENTER
DARCOM Intern Training Center - USALMC
Red River Army Depot
Texarkana, Texas 75501

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

FOREWORD

The research discussed in this report was accomplished as part of the Product/Production Engineering Graduate Program conducted jointly by DARCOM Intern Training Center and Texas A&M University. As such, the ideas, concepts and results herein presented are those of the author and do not necessarily reflect approval or acceptance by the Department of the Army.

This report has been reviewed and is approved for release. For further information on this project contact: Professor T. F. Howie, DRXMC-ITC-PPE, Red River Army Depot, Texarkana, Texas 75501.

Approved:

Professor T. F. HOWIE, P.E.
Chairman, Department of Product/Production Engineering

For the Commandant

JAMES L. ARNETT, Director
DARCOM Intern Training Center

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
DARCOM-ITC-02-08-76-215

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*
The Simulation and Analysis of Multi-Product Manufacturing Systems

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Alan S. Hartman

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Product/production Graduate Engineering Program, DARCOM Intern Training Center
Red River Army Depot, Texarkana, TX 75501

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Product/Production Graduate Engineering Program & Texas A&M University Graduate Center DARCOM-ITC-USALMC

12. REPORT DATE
April 1976

13. NUMBER OF PAGES
95

14. MONITORING AGENCY NAME & ADD[redacted]**olling Office)**

15. SECURITY CLASS. *(of this report)*

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*
Research performed by Alan S. Hartman under the supervision of Dr. J.W. Foster, Industrial Engineering Dept., Texas A&M University.

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*
Analytic Models
Models
Production Models
Simulation

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*
The development of a FORTRAN based simulation program which can be used to study and analyze any multi-product manufacturing system is the topic of this paper. A general one step Markov transition matrix is developed to generate each production unit's output. A detailed economic and statistical study of the manufacturing network is available as program options. The program was validated by using hypothetical conditions and assumptions on the Army's new system of munitions production SCAMP.

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Mr. T.F. Howie for his assistance in obtaining the necessary research trips and computer time. A special thanks is given to Dr. Joseph Foster for serving as committee chairman. Gratitude is also extended to Dr. Bart Childs, Mr. Patrick J. Hollifield, Mr. Charlie Clarkson, and Mr. B.C. Bowers for their help in computer programming and gathering of background information.

During the course of this work, the author was employed by the U.S. Army as a career intern in the DARCOM Product/Production Engineering Graduate Program. He is grateful to the U.S. Army for the opportunity to participate in this program.

The ideas, concepts, and results herein are those of the author and do not necessarily reflect approval or acceptance by the Department of the Army.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

The job of designing, optimizing and evaluating various production systems is becoming more difficult due to our increased knowledge of equipment and manufacturing processes. A means of analysis is needed that may be applied to both simple and complex manufacturing systems. Such a method of analysis must sufficiently relate effects of controllable production parameters upon dependent process parameters.

Parameters that are controllable through management decisions include choices among different equipment reliabilities and capabilities, parts and raw material inventory sizes, in-process buffer limitations, material handling techniques, production configurations, and general production policies. These production parameters have been shown to have a significant effect upon the dependent process parameters of systems reliability, systems availability, systems maintainability, production rates, production costs, and operating effectiveness.

Evaluation of production systems is currently accomplished by two basic methods of analysis; prototype testing and simulation. Prototype testing is usually an evaluation of a functioning, scaled model or limited application of a larger proposed system. Generally, this is a long and expensive method of evaluation. Computer simulation, however, is a technique that uses computer modeling as a symbolic representation for a definable system. It has proven to be a relatively easy, inexpensive and reliable technique of analysis, especially in the field of research. Simulated programs can also be easily manipulated to handle practically any type of change in a production system that normally would be impratical to try. Thus, a manufacturing system may be tested and optimized over a broad range of conditions with minimum effort. In comparing these two methods, simulation appears to be the most effective method for analysis. However, there needs to be a simulated program designed strictly for analysis of changes in the production process parameters of reliability, availability and maintainability and their effect upon the dependent output parameters.

Programs that are available for simulation such as DYNAMO, General Activity Simulation Program (GASP), SIMULATE, and General Purpose Systems Simulator (GPSS) are designed for simulating a wide range of problems. But their generalized approach and usage of different computer languages tend to make them difficult to apply. Also, in reporting the effect of parameter changes upon the system, desired statistical information on all operating characteristics may not be included. (1,3)

Simulation and Analysis of Manufacturing Systems (SAMS) is a newly developed program designed for the study and analysis of manufacturing systems. (3) Currently, it has been used to simulate and optimize the operating policy of a continuous single product manufacturing system. (1,3)
It is not capable of handling a multi-product manufacturing system.

The objective of this report is to develop SAMS further and to make it capable of simulating multi-product production systems. Chapter II is a literature review of SAMS logic in the single product simulation. Chapter III is a comparison of the single and multi-product systems with a brief discussion of changes that are necessary in the development of a multi-product simulating system. Chapter IV is a description of the Small Caliber Ammunition Modernization Program, SCAMP (1,3) and how theoretical changes could make SCAMP into a multi-product manufacturing system. Chapter V contains results on simulation runs from the SCAMP multi-product system. Chapter VI contains conclusions and recommendations.

Numbers in parenthesis refer to numbered references in the list of References.

CHAPTER II

LITERATURE SURVEY

Simulation and Analysis of Manufacturing Systems (SAMS) was designed by Snyder (3) to aid management in designing and evaluating any single product manufacturing system. Other general simulation programs that were available were not acceptable due to their generalized approach, their design for problems other than manufacturing systems and their lack of adequate statistical analysis. (1,3) Therefore, SAMS was designed to meet these specific deficiencies.

SAMS: Simulation Modeling

SAMS utilizes a time based, Monte Carlo simulation and the Markov transition matrix technique to generate the output of production units. (3) The basic concepts in the Monte Carlo simulations and the Markov processes are important because their limitations will be the same limitations as for any application of SAMS.

The time based Monte Carlo simulation assigns a range of numbers to all possible states of condition, both normal and abnormal, of a manufacturing system. Each state then receives a portion of the numbers that relate to its expected probability of occurence. A sort of random number is then generated, and the state of condition is determined by comparison with the assigned numbers. It is possible for these random numbers to follow a pattern of uniform, normal, Poission, or other distribution form. (2) With this method, a manufacturing system can be simulated under probable conditions of operation and its performance analyzed.

Usefulness of Monte Carlo simulations is generally limited in application by three main difficulties. First, only one set of system parameters may be specified for a simulation run. Second, it may be difficult to determine the number of trials that are necessary in achieving the desired level of confidence in results. Finally, the cost of Monte Carlo simulations can become costly in terms of computer time. These three difficulties also tend to interact with one another in such a way that decision tradeoffs between them must be made by the user. (6)

Any Markov system has two important features: first, the probabilities of future events are independent of past events (4); and secondly, all possible states of condition must be included in the Markov matrix of transition probabilities. (3) Markov transition matrices are used in stochastic systems to represent all probable states of condition, both normal and abnormal. Probability values listed in the matrix give the chances of a system transferring from one state of condition to another within a specified time interval.

Simulation of systems is accomplished by taking random numbers obtained through Monte Carlo simulation and determining probable states of condition through the Markov transition matrix. This simulated operation over a period of time may then be analyzed to determine how changes in controllable system parameters have affected any of the dependent characteristics of the system. In SAMS, the probability distributions of elements that make up a production unit are assumed to be exponentially distributed. Probabilities that are then calculated, representing probable states of condition for a production unit, will remain constant over time. The time interval for which transition probabilities are calculated must also be small enough so that the chance of changing to more than one state of condition within the time interval is negligibly small. (3)

SAMS: Network Description

There are two types of network symbols utilized by SAMS in depicting a material flow diagram for the production system. A square box indicates a production unit, and a circle represents a storage or buffer unit for material handling purposes. (3) SAMS requires the output of production units to enter a buffer before entering another production unit. However, this does not necessarily imply a one-to-one relationship.

All production process information will be described by the input data. This information includes data dealing with production units, buffers and pattern of material flow.

Data describing the production units include speed, mean time between failure (MTBF), mean time to repair (MTTR), and other information that is necessary for operating the unit and its elements. Buffer data includes the initial buffer size, upper and lower limits, MTBF, and MTTR. In describing the flow pattern of materials, the user must provide a matrix that states the relation between production units and buffers. Thus, SAMS allows any number of changes in the system's physical structure or operating characteristics to be manipulated by changing only the input data. (1)

SAMS simulates the output of a production unit by using the Markov transition matrix with random numbers generated through Monte Carlo simulation to determine the level of production for a time interval. Mathematical models that are capable of calculating transition probabilities for specified production models have been summarized by Snyder (3) from unpublished reports (5,6,7) prepared at the Intern Training Center by Texas A&M faculty.

SAMS includes the options of cost analysis and plotting. Data which must be included for the cost analysis option are purchase price, salvage value, expected life, operating and repair cost of all elements in the production unit, raw material cost, and quality of output. Information on production units, from which the finished product emerges, is furnished through the mainline input data. Information calculated by this option include total cost of production and cost per unit produced. The plotting option provides a graph for all buffer levels during production, at intervals specified for by the user. (3)

The functions of the mainline program are: to keep track of output, production units and buffer status; to start and stop parts of the simulation; and to control all program logic functions that will report on the operating characteristics of the system. After all input data on production units and buffers has been processed, SAMS will repeatedly simulate production for the number of time intervals in a day that the user has specified. Within each iteration, a subroutine is called which makes use of the Markov transition matrix to determine the state of condition for units and rates of production that were possible for that time interval. Buffer levels are then determined and checked based upon the production outputs of the different units. If either the upper or lower limits of a buffer are exceeded, appropriate production units are shut down. Next, the probability of a buffer failure or repair is simulated, and again appropriate actions are taken. This process is then repeated for the specified number of time intervals. For instance, four hundred and eighty iterations is equivalent to simulating the production process, at one minute intervals, for an eight hour shift. Specified options are then performed. This process is repeated for the specified number of days. After all specified time requirements have been processed, a statistical analysis is made which shows average production, average buffer size, production variance, availability of production units and buffers, variance of buffer levels, and production unit downtime caused by buffers. (1)

CHAPTER III

MULTI-PRODUCT SIMULATION

The single and the multi-product manufacturing systems are similar in structure. Both have production units that are capable of performing similar operations on different products. However, the multi-product manufacturing system is complicated due to problems in scheduling products for production, to changes in material flow patterns and to allocation of set-up times. Scheduling products for production is a new parameter to be considered in the optimization on a multi-product system. It is the objective of this report to make SAMS capable of simulating a broad range of conditions. Therefore, SAMS must be able to accept a change in scheduling of jobs for production units through input of such information. In a single product manufacturing system, material flow patterns are fixed characteristics; whereas, in the multi-product system, they are dependent upon job requirements. In-process and raw material inventory buffers will change as a production unit ceases work on one product and begins on another. Time for set-up procedures on the production unit must then be allowed within the simulation. These differences are included in the simulation for a SAMS multi-product manufacturing system.

Other changes in SAMS logic include decisions in simulation control and specific subroutine requirements. Specific production limits are assigned to production units for the different products. When these specified limits are reached, appropriate production units are shut down for the required set-up period while a change in jobs and material flow patterns occurs. Also, the need to design a specific subroutine for each production unit is changed to make one subroutine capable of calculating the three types of production models described by Snyder (3) and for other models to be included that are different in design.

The procedure and logic applications for a SAMS multi-product program will be the same as in the single product case except for a test to change jobs at the end of each production interval. Utilization of Monte Carlo simulation and Markov transition matrix techniques in generating the output of production units is again used. Basic concepts of Monte Carlo and Markov processes and their limitations also apply.

The procedure and logic for the mainline program are almost identical to that used in the single product program, except for specific changes which were previously noted. Appendix A contains a complete listing of the mainline program and subroutines. Figure 3.1 is a simplified flow chart for the operations and steps that the mainline program performs. One significant difference in the multi-product's mainline program is the test for change in jobs on different production units.

All production process information is presented in the input data, and a listing of the input information is given in Appendix B. Appendix C is an explanation of information terms used.

Upon execution of the program, SAMS multi-product simulation will perform the required number of intervals and days, processing all input information and performing specified options at appropriate times. Statistical analysis of the process is performed at the end of the simulation period.

The next chapter will describe a system that has been modeled as a single product manufacturing unit and how different conditions would make it a multi-product system.

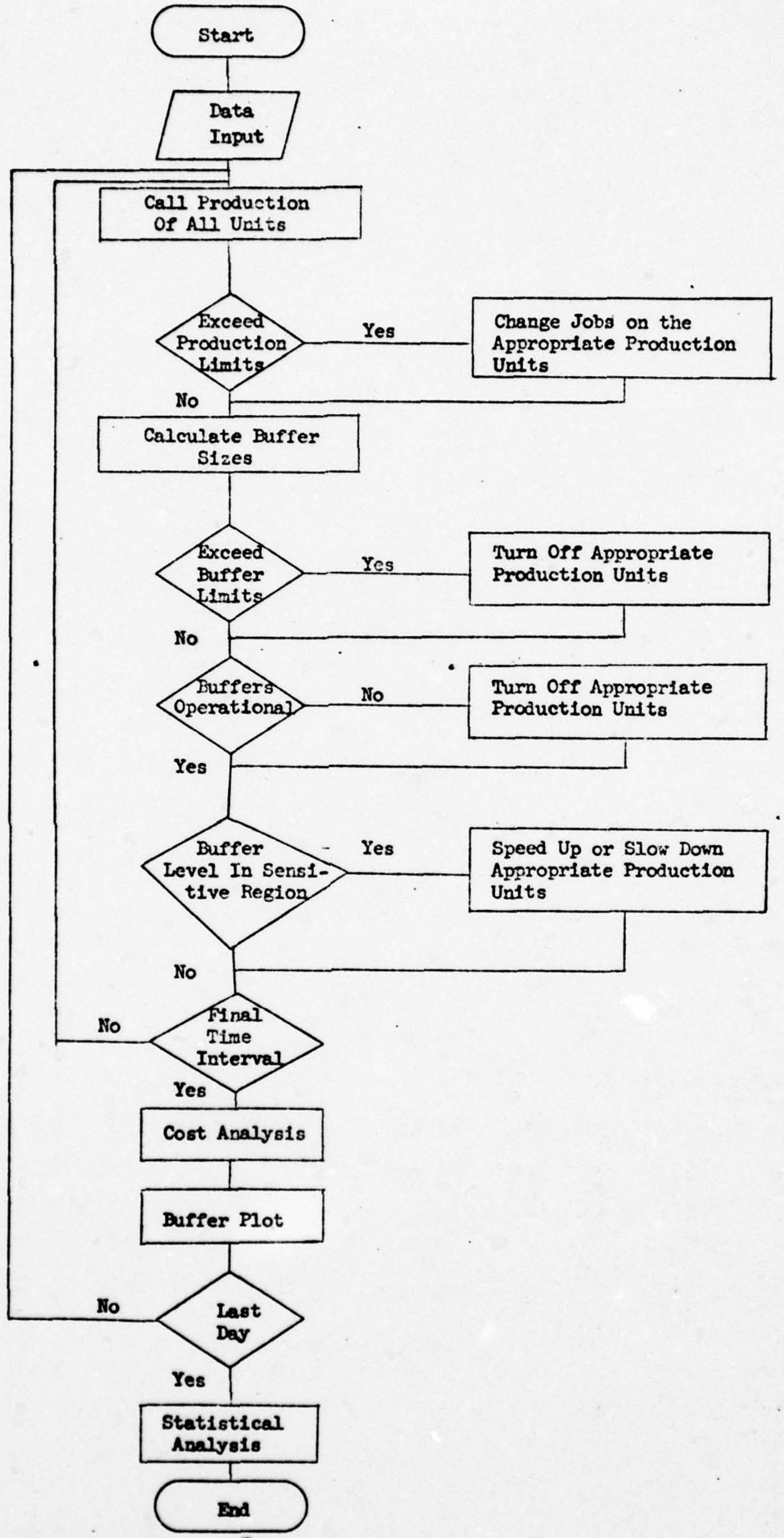


Figure 3.1 SAMS Multi-Product Flow Chart

CHAPTER IV

THE SCAMP MODULE A SYSTEM

The SCAMP Module A system is a continuous production system with captive components moving small caliber ammunition through production. (3) Four different types of production units, called submodules, are used in producing 5.56mm ammunition. Figure 4.1 is a basic logic configuration of the manufacturing process.

In the case submodule, brass cups are drawn, annealed, shaped, and trimmed to produce a finished case. At the primer insert submodule, finished cases are inspected, and gaged primers are inserted. At a bullet submodule, bullet projectiles are manufactured and fed into the load and assemble submodule. The load and assemble submodule takes the cases, adds propellant and joins the bullet projectiles to complete a finished round.

For example, production within the case submodule is performed by a series of eighteen rotary turrets, each with twenty-four tool stations. This could be considered as twenty-four lines of production. Following the turrets is a series of twenty-eight single work stations, see Figure 4.2. Tool failure at any rotary station will cause the production

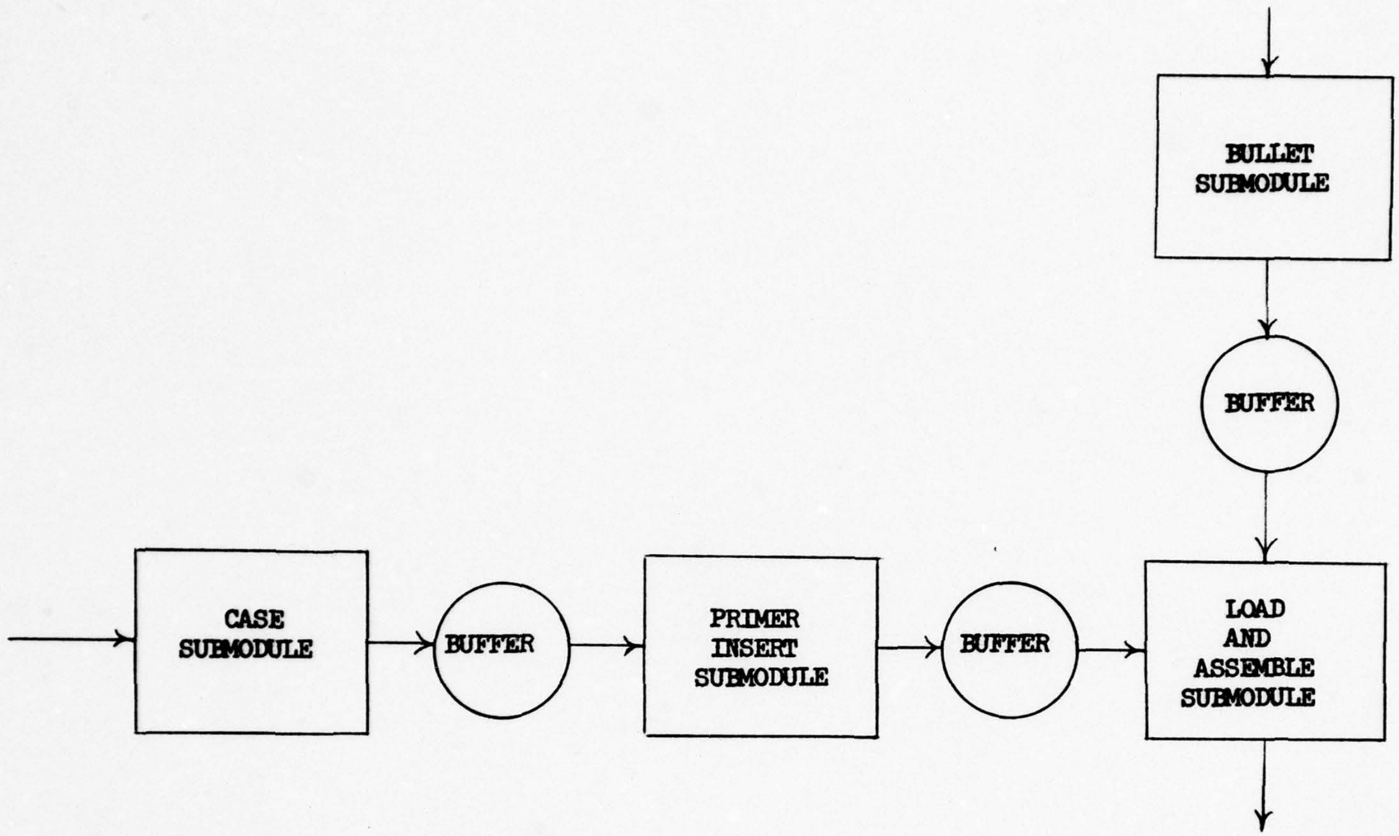


Figure 4.1 Basic Configuration of SCAMP Module A

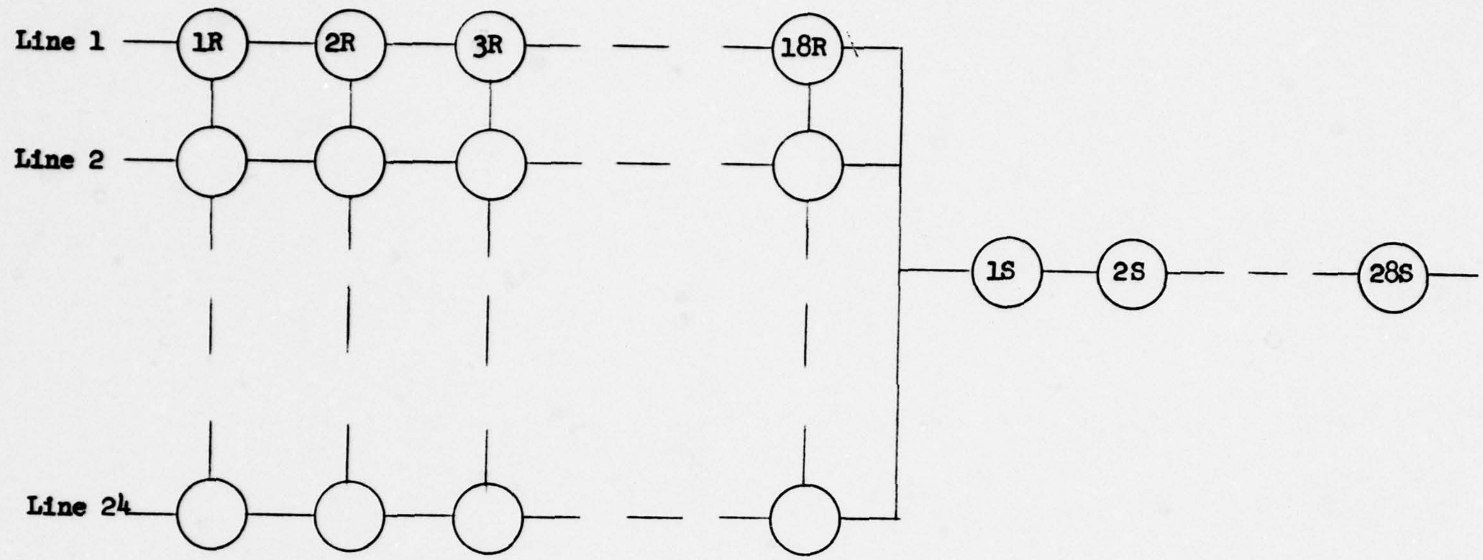


Figure 4.2 Basic Configuration for SCAMP's Module A Case Submodule

of that line to be lost; whereas failure of any single, serial work station will automatically cause the entire submodule to shut down and await repair. Similar use of rotary and serial elements is made for production configuration of the other submodules. (1)

One of Snyder's basic assumptions for this system was that if a single, serial work station failed, then the mean time to repair of this serial failure would always be greater than the mean time to repair any rotary tool failures that had occurred prior to the serial failure. Recent data has shown that this is not a valid assumption. Only a limited number of rotary failures can be repaired in the time that it takes to repair a serial failure. If more rotary failures have occurred than are possible to repair during the time it takes to repair one serial failure, time must be included to repair the additional rotary failures. This change in programming was made by Mr. Pat Hollifield, former instructor at the Intern Training Center (ITC), and Mr. Charlie Clarkson, former SCAMP coordinator for research projects at ITC.

In describing how Markov processes are used in this system, an operating policy for the submodules must be assumed such that after a specific number of rotary failures, the submodule will shut down for repair. This policy may then be used in calculation of probabilities for the Markov transition matrix. For example, if the policy is to shut down the case submodule after six tool failures or after any single serial failure, then the Markov matrix must account for changes in the rotary and the serial repair probabilities. Table 4.1 lists the set of mutually exclusive states for the given operating policy. The design for the Markov matrix of transition probability is shown in Table 4.2. Calculation for transition probabilities in areas I and II are shown in Table 4.3.

Additional products that could ideally be manufactured by some of this equipment would include high explosive (HE), high explosive incendary (HEI) and tracer ammunition rounds. The bullet submodule could feed HE, HEI or tracer explosive mixtures from bulk hoppers, measure the mixture and then force feed the mixture into a projectile body, depending on the specific type of requirements. This submodule could manufacture all parts required or take completed items from private suppliers to be a load and test submodule. In either case, the completed projectiles are fed into buffers for use in the load and assemble submodule.

In changing SAMS to simulate this multi-product system, certain assumptions were made in the basic operating characteristics of the system. First, probabilities of failure in the submodules' components were assumed constant over time regardless of changes in product production. Second, production rates were also assumed the same for all products, and finally, the operating policy for submodules was also assumed the same for all products. A system configuration for this multi-product concept is shown in Figure 4.3. The next chapter will present results on simulation runs of this multi-product system.

STATE NUMBER	DESCRIPTION	PRODUCTION RATE
1	0 Rotary Failures	1200 rds/min
2	1 Rotary Failure	1150 rds/min
3	2 Rotary Failures	1100 rds/min
4	3 Rotary Failures	1050 rds/min
5	4 Rotary Failures	1000 rds/min
6	5 Rotary Failures	950 rds/min
7	6 Rotary Failures	900 rds/min
8	Serial Failure w/ No Previous Rotary Failures	0 rds/min
9	Serial Failure w/ 1 Previous Rotary Failure	0 rds/min
10	Serial Failure w/ 2 Previous Rotary Failures	0 rds/min
11	Serial Failure w/ 3 Previous Rotary Failures	0 rds/min
12	Serial Failure w/ 4 Previous Rotary Failures	0 rds/min
13	Serial Failure w/ 5 Previous Rotary Failures	0 rds/min

Table 4.1 Possible States of Condition for SCAMP's Case Submodule

	1	2	3	4	5	6	7	8	9	10	11	12	13
1							A	p^*	0	0	0	0	0
2	0		A R E A				R	0	p^*	0	0	0	0
3	0	0			I		E	0	0	p^*	0	0	0
4	0	0	0				A	0	0	0	p^*	0	0
5	0	0	0	0			II	0	0	0	0	p^*	0
6	0	0	0	0	0			0	0	0	0	0	p^*
7	r_7^*	0	0	0	0	0	$1-r_7^*$	0	0	0	0	0	0
8	r_8^*	0	0	0	0	0	0	$1-r_8^*$	0	0	0	0	0
9	r_9^*	0	0	0	0	0	0	0	$1-r_9^*$	0	0	0	0
10	r_{10}^*	0	0	0	0	0	0	0	0	$1-r_{10}^*$	0	0	0
11	r_{11}^*	0	0	0	0	0	0	0	0	0	$1-r_{11}^*$	0	0
12	r_{12}^*	0	0	0	0	0	0	0	0	0	0	$1-r_{12}^*$	0
13	r_{13}^*	0	0	0	0	0	0	0	0	0	0	0	$1-r_{13}^*$

Table 4.2 Markov Transition Matrix for SCAMP's Case Submodule

<u>AREA I</u> $(1 \leq i \leq n)$

$j \leq i$ $\quad p_{ij} = 0$

$i \leq j \leq n$ $\quad p_{ij} = \binom{24-i+1}{j-1} p^{j-1} (1-p)^{24-j+1} (1-p^*)$

<u>AREA II</u> $(1 \leq i \leq n)$

$j = n+1$ $\quad p_{ij} = \sum_{k=n-i+1}^{24-i+1} \binom{24-i+1}{k} p^{i-1} (1-p)^{24-i+1} (1-p^*)$

$j = n+i+1$ $\quad p_{ij} = p^*$

where:

$p = Pr(\text{rotary failure in } \Delta t)$

$p^* = Pr(\text{serial failure in } \Delta t)$

for calculation of r_i^* where $(7 \leq i \leq 13)$:

$r_7^* = Pr(6MTTR_r + 6JOG + TSS)$

$r_8^* = Pr(MTTR_s + TSS)$

$r_9^* = Pr(Max(MTTR_s, MTTR_r + JOG) + TSS)$

$r_{10}^* = Pr(Max(MTTR_s, 2MTTR_r + 2JOG) + TSS)$

$r_{11}^* = Pr(Max(MTTR_s, 3MTTR_r + 3JOG) + TSS)$

$r_{12}^* = Pr(Max(MTTR_s, 4MTTR_r + 4JOG) + TSS)$

$r_{13}^* = Pr(Max(MTTR_s, 5MTTR_r = 5JOG) + TSS)$

where:

$MTTR_s$ = mean time to repair serial failure

$MTTR_r$ = mean time to repair a rotary failure

JOG = time required to turn turret for tool change

TSS = time required to start and stop submodule

Table 4.3 Calculation for
Markov Transition Matrix (Table 4.2)

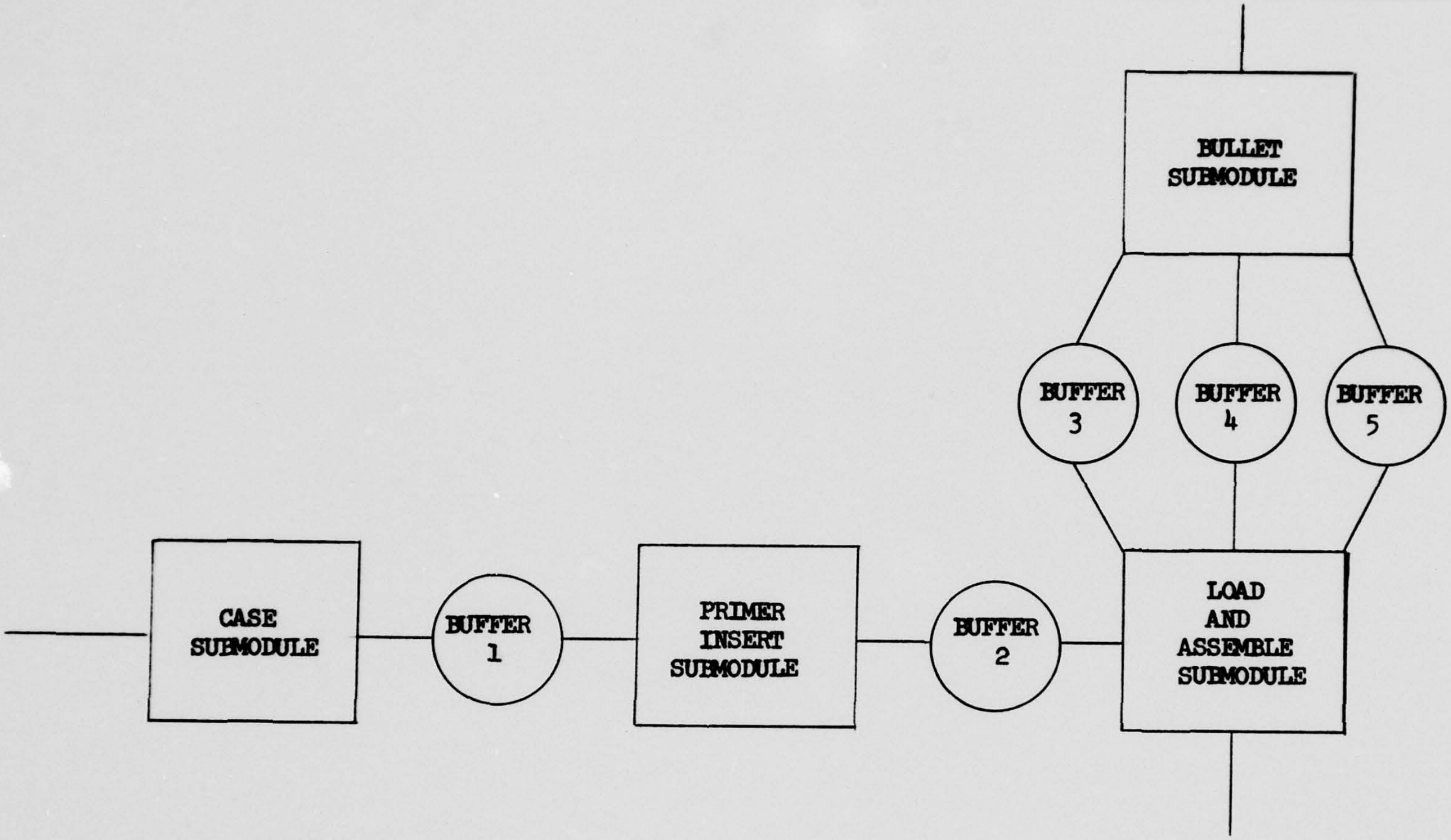


Figure 4.3 Basic Configuration of SCAMP's Multi-Product System

CHAPTER V

TESTING

In validating SAMS multi-product programming, hypothetical conditions were assumed that required the SCAMP system to produce one minute's production and then shut down appropriate submodules for specified set-up periods. Minute by minute status printouts were made and analyzed to validate that the simulation was performing satisfactorly. Different conditions were assumed, for example purposes, that varied buffer sizes, frequency of job changes and different set-up times. A common comparison time period of one week was selected.

Differences in the three controllable production parameters consisted of two levels in each condition. Large and small buffer levels reflect the different upper limit values. A small buffer case consisted of an upper limit of twenty-five thousand units, while a large buffer case was represented by fifty thousand units upper limit. This meant that production units feeding these buffers would be shut off when their limits were exceeded. Differences in job frequency dealt with a low frequency situation, approximately one change in jobs per day, while a high frequency case consisted of approximately three changes in jobs a day. Short and long set-up times were listed as ten minutes and ninety minutes between jobs, respectively. These times were set equal for all machines that required a change in jobs, although they could have been specified differently. Using these different conditions, six runs were made for comparison. The same random number streams were used in all runs so that operating characteristics of production units would coincide and effects of controllable production parameters upon the system could be observed. The six runs previously mentioned were:

1. Small Buffer Limits; High Frequency in Jobs; Short Set-up Time
2. Small Buffer Limits; Low Frequency in Jobs; Short Set-up Time
3. Large Buffer Limits; Low Frequency in Jobs; Short Set-up Time
4. Large Buffer Limits; High Frequency in Jobs; Short Set-up Time
5. Small Buffer Limits; High Frequency in Jobs; Long Set-up Time
6. Large Buffer Limits; High Frequency in Jobs; Long Set-up Time

Appendix D is a listing of the statistical analysis on these different runs and Table 5.1 summarizes parts of these reports.

Some of the differences between runs were as expected when controllable operating parameters were changed. For example, a lengthening of set-up time was expected to increase the cost of production. This was verified by noticing

	RUNS 1	2	3	4	5	6
Cost Per Unit Analysis	0.04	0.04	0.04	0.04	0.054	0.054
Average Unit Production/Interval						
Unit 3	196	352	367	362	279	281
Unit 4	187	342	357	356	283	288
Production Unit Availability						
Unit 3	20	36	37	37	29	29
Unit 4	23	42	43	43	35	35
Unit Downtime Due to Buffers						
Unit 3	60	32	29	26	16	13
Unit 4	49	11	8	6	9	7

Table 5.1 Partial Summary of Information Contained in Output Reports

the difference between unit costs for the different production runs involving short and long set-up times. Another difference was noted in average rate of unit production per time interval, production availability and unit downtime due to buffers between run one and runs two through six. These differences would warrant further analysis, under normal circumstances, especially since Snyder noted that changes in buffer limits were insensitive to systems performance for the single product case. Changes in submodules and buffer reliability, as well as probabilities of failures, can be made and analyzed for comparison of operating characteristics on different production systems. Similar procedures of optimization for controllable production parameters may then be applied, as Snyder and Chu have already done for the single product case.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

The objective of this report was to extend the SAMS single product manufacturing system to simulate multi-product production systems. Chapter II described functioning characteristics of SAMS, in the single product state. Chapter III compared single and multi-product systems and included a discussion of changes that were necessary in developing the multi-product simulation system. Chapter IV contained a brief description of a single product production problem that has been studied by Snyder (3) and by Chu (1) and described theoretical changes that made it a multi-product problem. Chapter V compared and listed the results of six different simulation runs on SAMS multi-product manufacturing system.

The example, SCAMP simulated as a multi-product manufacturing system, showed the functioning capabilities of SAMS in a multi-product system, yet there are additional changes in the programming that would make SAMS capable of handling even more complex multi-product systems. For example, it is not always desirable or possible to have all products produced at equivalent rates. Some products may take longer to have similar operations performed by the same equipment. Also, as production units change jobs, the probability of machine failure does not remain constant.

Methods for analysis and optimization, as used by Snyder (3) and Chu (1), may be used in making SAMS a useful and valuable tool in the evaluation and analysis of varied production systems.

APPENDIXES

APPENDIX A

```
C MAINLINE PROGRAM
C
C
      REAL LOLIM(10),LIMIT(10,3)
      DIMENSION A(3,10,10),MFAIL(3,10,10),KN(10,3),SETUP(10,3)
      DIMENSION          IFAIL(481,10),PROD(481,10),ISTAT(481,10)
      DIMENSION BUFF(481,10),POUT(10,20), IBFAL(481,10),ULIM(10)
      DIMENSION BPROB(10),ADJUS(10),KFAST(10),KSLOW(10),TOTAL(10)
      DIMENSION SEN(10),AVAM(10),NU(25),MDES(40),TPROD(10),NX(25)
      DIMENSION TBUFF(10),TDOWN(10),AVAB(10),TFAIL(10)
      DIMENSION STDEP(60,10),AVAMB(60,10),TOTPC(60),SETUPS(10)
      DIMENSION DOWN(60,10),TAVG(10),TAVAM(10),TSTOM(10),TTDON(10)
      DIMENSION AVGPP(60,10),STDEM(60,10),AVAMM(60,10),AVGBU(60,10)
      DIMENSION XX(10),YY(10),BPROC(10),RMTTR(10)
      DIMENSION TAVGB(10),TAVAB(10),TSTDB(10),ISEED(10)
      DIMENSION BMAT(21,23,10),RATS(10),IATS(2,10)
      INTEGER SMINDX(10),SMTYPE(10)
C
      DATA A/300 *0./,BMAT/4830*0./,POUT/200*0./,
     *MFAIL/300*0/ ,TOTAL/10*0./,LIMIT/30 *0./
      COMMON IN,IO,IP,KN,NJ,ADJUS,NRBMAT,NCBMAT,RATS,IATS
      DATA NPT/'-'/
C CALCULATION FOR EXPONENTIAL PROBABILITY
      PROB( T,X)=1.0 - 1.0 / EXP(1.0/ T*X)
C  INDICATE THE NUMBER OF ROWS AND COLUMNS IN BMAT, SCAMP ONLY
      NRBMAT=21
      NCBMAT=23
C
C  IN - CONTROL INPUT FOR READER
C  IO - CONTROL OUTPUT FOR PRINTER
C  IP - CONTROL OUTPUT FOR PUNCH
C
      IN=5
      IO=6
      IP= 7
      KT=0
      WRITE(IO,405)
  405 FORMAT('1',///////// )
C
C  READS AND WRITES FIRST 25 DATA CARDS,  ALSO INITILIZES VECTORS NU & NX
C
      DO 45 IM=1,25
      NU(IM)=IM
      NX(IM)=NPT
      READ(IN,400) (MDES(L),L=1,40)
  400 FORMAT(40A2)
      WRITE(IO,400) (MDES(L),L=1,40)
   45 CONTINUE
C
C INPUT OF PROCESS PARAMETERS AND INITAL CONDITIONS
C
C  N   NUMBER OF PRODUCTION UNITS IN SYSTEM
C  NB  NUMBER OF BUFFERS IN SYSTEM
C  NJ  NUMBER OF JOBS SYSTEM IS TO PRODUCE
C INTVL NUMBER OF TIME INTERVALS PLUS ONE THAT ARE TO BE SIMULATED
C ISKIP PARAMETER FOR COST ANALYSIS IF DESIRED
C NINT NUMBER OF TIME INTERVALS BETWEEN CURRENT STATUS PRINTOUTS ON PRODUCTION
```

```
C       UNITS AND BUFFERS
C NPROD NUMBER OF TIME INTERVALS BETWEEN SUMMARY PRINTOUTS OF PRODUCTION UNITS
C       AND BUFFER OPERATIONS TO PRESENT
C NDAYS NUMBER OF DAYS TO RUN SIMULATION
C JPLOT PARAMETER OF PLOTTING ALL BUFFER LEVELS OVER TIME IF DESIRED
C NPRNT NUMBER OF TIME INTERVALS BETWEEN POINTS ON THE PLOTS OF.BUFFER LEVELS
C NPLOT FREQUENCY IN DAYS OF PLOTS IN BUFFER LEVELS OVER TIME
C IX  SEED FOR BUFFER RANDOM NUMBER GENERATION
      READ(IN,100) N,NB,NJ,INTVL,JSKIP,NINT,NPROD,NDAYS,JPLOT,NPRNT,     SAMS0035
     *NPLOT,IX                                                           SAMS0036
  100 FORMAT(16I5)                                                       SAMS0037
C
C  BUILD THE MARKOV MATRIX FOR N PRODUCTION UNITS
C
      DO 31 J=1,N                                                        SAMS0038
      CALL MKVBLD (SMINDX(J),BMAT,POUT,SMTYPE(J),LIMIT   ,ISEED(J))      SAMS0039
   31 CONTINUE                                                           SAMS0040
C
C A(K,I,J)  LOGIC FLOW FOR MATERIAL BY JOB (K) THROUGH PRODUCTION UNITS (J) AND
C           BUFFERS (I).
C ISTAT(I,J)  MATRIX STATE FOR PRODUCTION UNIT (J) AT END OF TIME INTERVAL (I).
C NFAIL(K,K,J)  LOGIC CONTROL FOR PRODUCTION UNITS (J) BY JOBS (K) DEPENDING ON
C                  CONDITION OF BUFFERS (I).
C BUFF(I,J)  NUMBER OF UNITS IN BUFFER (J) AT THE END OF TIME INTERVAL (I).
C ULIM(I)  UPPER LIMIT OF BUFFER (I).
C LOLIM(I)  LOWER LIMIT OF BUFFER (I).
C BPROB(I) PROBABILITY OF OPERATION FOR BUFFER (I).
C BMTIR(I)  MEAN TIME TO REPAIR ON BUFFER (I).
C ADJUS(J)  PERCENTAGE PRODUCTION UNIT (J) CAN BE SLOWED DOWN OR SPED UP.
C SEN(I)  SENSITIVITY FOR SLOWING DOWN OR SPEEDING UP PRODUCTION UNITS ON THE
C         BASIS OF BUFFER LEVELS (I).
C SETUP(J,K)  TIME REQUIRED TO SET-UP PRODUCTION UNITS (I) FOR JOBS (K).
C
      DO 32 K=1,NJ                                                       SAMS0041
      DO 30 J=1,NB                                                       SAMS0042
      READ(IN,200) (A(K,J,I),I=1,N)                                      SAMS0043
  200 FORMAT(10F8.0)                                                     SAMS0044
   30 CONTINUE                                                           SAMS0045
   32 CONTINUE                                                           SAMS0046
      READ(IN,300) (ISTAT(I,L),L=1,N)                                    SAMS0047
  300 FORMAT(40I2)                                                       SAMS0048
      DO 34 K=1,NJ                                                       SAMS0049
      DO 50 JA=1,NB                                                      SAMS0050
      READ(IN,300) (NFAIL(K,JA,JB),JB=1,N)                               SAMS0051
   50 CONTINUE                                                           SAMS0052
   34 CONTINUE                                                           SAMS0053
      READ(IN,200) (BUFF(1,J),J=1,NB)                                    SAMS0054
      READ(IN,200) (ULIM(I),I=1,NB)                                      SAMS0055
      READ(IN,200) (LOLIM(I),I=1,NB)                                     SAMS0056
      READ(IN,200) (BPROB(K),K=1,NB)                                     SAMS0057
      READ(IN,200) (BMTIR(K),K=1,NB)                                     SAMS0058
      READ(IN,200) (ADJUS(L),L=1,N)                                      SAMS0059
      READ(IN,200) (SEN(I),I=1,NB)                                       SAMS0060
      DO 35 J=1,NJ                                                       SAMS0061
      READ(IN,200)  (SETUP(I,J),I=1,N)                                   SAMS0062
   35 CONTINUE                                                           SAMS0063
C
C PRINTING INITIAL CONDITIONS
```

```
C
      WRITE(IU,434)
  434 FORMAT('1',/////////,10X,'INITIAL CONDITIONS',/,10X,  18('-'),
     1            //// ,01X,' UNIT    ',                         'ENTERING
     2 SHUT  ADJUSTME IT  ',/,1X,
     3                     'NUMBER   STATE    DOWN   (PERCENT)  ',/,01X,'----
     4--  --------  ----  ----------     ')
      DO 55 L=1,N
      IFAIL(2,L)=0
      R8=100.*ADJUS(L)
      WRITE(IU,313) L,ISTAT(1,L),IFAIL(2,L),RB
  313 FORMAT(1H ,2X,I2,06X,I2,06X,I2,06X,F5.0)
   55 CONTINUE
      DO 47 L=1,NB
      BPROC(L)=PROB(BMTTR(L))
   47 CONTINUE
      WRITE(IU,212)
  212 FORMAT(///        ,02X,'BUFFERS ',/,02X, 7('-')     ,/,09X,'INITIAL
     1 LOWER   UPPER   PROBABILITY    SENSITIVITY  MEAN TIME   PROBABILI
     4TY'                                               ,/,01X,'NUMBER   SI
     2ZE     LIMIT   LIMIT   OF OPERATION   (PERCENT)    TO REPAIR   OF
     5REPAIR'                                                   ,/,01X,'---
     3---  -------  ------- -------  ------------   -----------  -------
     6--   -----------')
      DO 65 L=1,NB
      RA=100.*SEN(L)
      WRITE(IO,404) L,BUFF(1,L),LOLIM(L),ULIM(L),BPROB(L),RA,BMTTR(L) ,
     *BPROC(L)
  404 FORMAT(1H ,02X,I2,5X,F6.0,3X,F5.0,2X,F6.0,06X,F5.3,09X,F5.0,7X,F7.
     14,5X,F5.3)
   65 CONTINUE
      DO 9 K=1,NJ
      WRITE(IO,609) K,(NU(I),I=1,N)
  609 FORMAT('1',////////// ,15X,'DEPENDENCE MATRIX ',/,15X, 17('-'),//
     *,20X,'JOB ',I2,//,20X,'PRODUCTION UNIT',/,16X,15I5)
      WRITE(IO,612)
  612 FORMAT(    12X,'BUFFER ')
      DO 56 I=1,NB
      WRITE(IO,614) I,(MFAIL(K,I,KL),KL=1,N)
  614 FORMAT(/,11X,15I5)
   56 CONTINUE
      WRITE(IO,611)
  611 FORMAT(////    ,20X,'CODE',/,20X,'----',//,10X, '0   PRODUCTION UN
     1IT AND BUFFER NOT DIRECTLY CONNECTED  ',/,10X,'1   BUFFER INPUTS T
     2O PRODUCTION UNIT ',/,10X,'2   PRODUCTION UNIT INPUTS TO BUFFER')
    9 CONTINUE
      DO 7 K=1,NJ
      WRITE(IO,745) K,(NU(I),I=1,N)
  745 FORMAT('1',///////,13X,'BUFFER AND  UNIT  DISTRIBUTION TABLE' ,//,
     *13X,36('-'),///,27X,'JOB ',I2,
     +                 ///,21X,'PRODUCTION UNIT',/,14X,10I7)
      WRITE(IO,746)
  746 FORMAT(10X,'BUFFER')
      DO 67 L=1,NB
      WRITE(IO,743) L,(A(K,L,LL),LL=1,N)
  743 FORMAT (/,12X,I2,02X,13F7.3)
   67 CONTINUE
```

```
      WRITE(10,742)                                                       SAMS0120
  742 FORMAT(/// ,25X,'CODE'         ,//,10X,'ZERO',11X,'UNIT    AND BUSAMS0121
     1FFER NOT DIRECTLY CONNECTED',/,10X,'POSITIVE NO.',03X,'PROPORTION SAMS0122
     2OF  UNIT   PRODUCTION ADDED TO BUFFER',/,10X,'NEGATIVE NO.',03X,'PSAMS0123
     3ROPORTION OF  UNIT   PRODUCTION TAKEN FROM BUFFER ')            SAMS0124
    7 CONTINUE                                                        SAMS0125
C
C
C         LOOPING SIMULATION THROUGH DESIRED NUMBER OF DAYS
C
C
      DO 1000 NDAY=1,NDAYS                                            SAMS0126
      KOUNT=1                                                         SAMS0127
      KOTT=0                                                          SAMS0128
      KOT=0                                                           SAMS0129
      DO 10 I=1,INTVL                                                 SAMS0130
      DO 20 L=1,N                                                     SAMS0131
      IFAIL(I,L)=0                                                    SAMS0132
   20 CONTINUE                                                        SAMS0133
      DO 22 L=1,NB                                                    SAMS0134
      IBFAL(I,L)=0                                                    SAMS0135
   22 CONTINUE                                                        SAMS0136
   10 CONTINUE                                                        SAMS0137
      DO 49 J=1,N                                                     SAMS0138
      ISTAT(2,J) = 0                                                  SAMS0139
      KFAST(J)=0                                                      SAMS0140
      KSLOW(J)=0                                                      SAMS0141
   49 CONTINUE                                                        SAMS0142
C
C         LOOPING SIMULATION THROUGH TIME INTEVALS
C
      DO 60 J=2,INTVL                                                 SAMS0143
C
C         CALLING THE UNITS FOR THEIR PRODUCTION
C
      DO 15 L=1,N                                                     SAMS0144
C NORMALLY USE SMPROD, BUT FOR THIS SCAMP EXAMPLE MUST USE SAMPRD
C     CALL SMPROD (SMIDX(L),BMAT,XY(L),ISTAT(J,L),IFAIL(J,L),PROD(J,L),
C    *ISTAT(J-1,L),SMTYPE(L),LIMIT(L),POUT,KFAST(L),KSLOW(L),ISEED(L))
      CALL SAMPRD (SMIDX(L),BMAT,RBMAT,VCBMAT,RATS,IATS,              SAMS0145
     +             ISTAT(J-1,L),ISTAT(J,L),PROD(J,L),KFAST(L),        SAMS0146
     +             KSLOW(L),ADJUS(L),IFAIL(J,L),XX(L))                SAMS0147
   15 CONTINUE                                                        SAMS0148
      DO 21 L=1,N                                                     SAMS0149
      KFAST(L)=0                                                      SAMS0150
      KSLOW(L)=0                                                      SAMS0151
      KSUB=KN(L,1)                                                    SAMS0152
      TOT=TOTAL(L) + PROD(J,L)                                        SAMS0153
      IF(TOT .LE. LIMIT(L,KSUB)) GO TO 24                             SAMS0154
      PROD(J,L)=LIMIT(L,KSUB) - TOTAL(L)                              SAMS0155
   24 CONTINUE                                                        SAMS0156
   21 CONTINUE                                                        SAMS0157
   23 CONTINUE                                                        SAMS0158
C
C         CALCULATING THE NEW BUFFER SIZES
C
      DO 70 L=1,NB                                                    SAMS0159
      BITPU=0.                                                        SAMS0160
```

QUALITY PRACTICABLE
D TO DDC

```
      WRITE(IO,742)                                                     SAMS0120
  742 FORMAT(///  ,25X,'CODE'          ,//,10X,'ZERO',11X,'UNIT    AND BUSAMS0121
     1FFER NOT DIRECTLY CONNECTED',/,10X,'POSITIVE NO.',03X,'PROPORTION SAMS0122
     2OF  UNIT   PRODUCTION ADDED TO BUFFER',/,10X,'NEGATIVE NO.',03X,'PSAMS0123
     3ROPORTION OF  UNIT   PRODUCTION TAKEN FROM BUFFER ')              SAMS0124
    7 CONTINUE                                                          SAMS0125
C
C
C        LOOPING SIMULATION THROUGH DESIRED NUMBER OF DAYS
C
C
      DO 1000 NDAY=1,NDAYS                                              SAMS0126
      KOUNT=1                                                           SAMS0127
      KOUT=0                                                            SAMS0128
      KOT=0                                                             SAMS0129
      DO 10 I=1,INTVL                                                   SAMS0130
      DO 20 L=1,N                                                       SAMS0131
      IFAIL(I,L)=0                                                      SAMS0132
   20 CONTINUE                                                          SAMS0133
      DO 22 L=1,NB                                                      SAMS0134
      IBFAL(I,L)=0                                                      SAMS0135
   22 CONTINUE                                                          SAMS0136
   10 CONTINUE                                                          SAMS0137
      DO 49 J=1,N                                                       SAMS0138
      ISTAT(2,J) = 0                                                    SAMS0139
      KFAST(J)=0                                                        SAMS0140
      KSLOW(J)=0                                                        SAMS0141
   49 CONTINUE                                                          SAMS0142
C
C        LOOPING SIMULATION THROUGH TIME INTEVALS
C
      DO 60 J=2,INTVL                                                   SAMS0143
C
C        CALLING THE UNITS FOR THEIR PRODUCTION
C
      DO 15 L=1,N                                                       SAMS0144
C NORMALLY USE SMPROD, BUT FOR THIS SCAMP EXAMPLE MUST USE SAMPRD
C     CALL SMPROD (SMINDX(L),BMAT,XX(L),ISTAT(J,L),IFAIL(J,L),PROD(J,L),
C    *ISTAT(J-1,L),SMTYPE(L),LIMIT(L),POUT,KFAST(L),KSLOW(L),ISEED(L))
      CALL SAMPRD (SMINDX(L),BMAT,NRBMAT,NCBMAT,RATS,IATS,              SAMS0145
     +             ISTAT(J-1,L),ISTAT(J,L),PROD(J,L),KFAST(L),          SAMS0146
     +             KSLOW(L),ADJUS(L),IFAIL(J,L),XX(L))                  SAMS0147
   15 CONTINUE                                                          SAMS0148
      DO 21 L=1,N                                                       SAMS0149
      KFAST(L)=0                                                        SAMS0150
      KSLOW(L)=0                                                        SAMS0151
      KSUB=KN(L,1)                                                      SAMS0152
      TOT=TOTAL(L) + PROD(J,L)                                          SAMS0153
      IF(TOT .LE. LIMIT(L,KSUB)) GO TO 24                               SAMS0154
      PROD(J,L)=LIMIT(L,KSUB) - TOTAL(L)                                SAMS0155
   24 CONTINUE                                                          SAMS0156
   21 CONTINUE                                                          SAMS0157
   23 CONTINUE                                                          SAMS0158
C
C        CALCULATING THE NEW BUFFER SIZES
C
      DO 70 L=1,NB                                                      SAMS0159
      BITPU=0.                                                          SAMS0160
```

```
      PUITB=0.                                                          SAMS0161
      TA=0.                                                             SAMS0162
      DO 80 LA=1,N                                                      SAMS0163
      KSUB=KN(LA,1)                                                     SAMS0164
      IF (MFAIL(KSUB,L,LA)-1) 80,82,83                                  SAMS0165
   82 B2=PROD(J,LA)*A(KSUB,L,LA)                                        SAMS0166
      TA=TA+A(KSUB,L,LA)                                                SAMS0167
      BITPU=BITPU+B2                                                    SAMS0168
      GO TO 80                                                          SAMS0169
   83 B2=PROD(J,LA)*A(KSUB,L,LA)                                        SAMS0170
      PUITB=PUITB+B2                                                    SAMS0171
   80 CONTINUE                                                          SAMS0172
      BUFF(J,L)=BUFF(J-1 ,L)+PUITB                                      SAMS0173
      IF(BUFF(J,L) .GE. ABS(BITPU)) GO TO 71                            SAMS0174
      DO 84 LA=1,N                                                      SAMS0175
      IF(MFAIL(KSUB,L,LA)-1) 84,85,84                                   SAMS0176
   85 PROD(J,LA)=BUFF(J,L)/(ABS(TA))                                    SAMS0177
   84 CONTINUE                                                          SAMS0178
      GO TO 23                                                          SAMS0179
   71 BUFF(J,L)=BUFF(J,L)+BITPU                                         SAMS0180
   70 CONTINUE                                                          SAMS0181
      DO 81 LA=1,N                                                      SAMS0182
      TOTAL(LA)=TOTAL(LA)+PROD(J,LA)                                    SAMS0183
   81 CONTINUE                                                          SAMS0184
C     TEST FOR CHANGE IN JOBS ON BASIS OF PRODUCTION COMPLETED AGAINST PRESET LIMIT
      DO 94 JK=1,N                                                      SAMS0185
      KSUB=KN(JK,1)                                                     SAMS0186
      IF (LIMIT(JK,KSUB)-TOTAL(JK))53,53,94                             SAMS0187
   53 CALL CHGJOB(J,SETUP,IFAIL,JK)                                     SAMS0188
      TOTAL(JK    )=0.                                                  SAMS0189
   94 CONTINUE                                                          SAMS0190
C
C          COMPARATING THE BUFFER SIZES TO THEIR LIMITS
C
      DO 90 JP=1,IB                                                     SAMS0191
      IF(BUFF(J,JP)-LLLIM(JP)) 91,91,96                                 SAMS0192
   91 DO 92 JR=1,N                                                      SAMS0193
      IF(IFAIL(J+1,JR) .EQ. 2) GO TO 92                                 SAMS0194
      KSUB=KN(JR,1)                                                     SAMS0195
      IF(MFAIL(KSUB,JP,JR) .EQ. 1         ) GO TO 95                    SAMS0196
      GO TO 92                                                          SAMS0197
   95 IFAIL(J+1,JR)=1                                                   SAMS0198
   92 CONTINUE                                                          SAMS0199
   96 IF(BUFF(J,JP)-ULIM(JP)) 90,90,97                                  SAMS0200
   97 DO 98 JS=1,N                                                      SAMS0201
      IF(IFAIL(J+1,JS) .EQ. 2) GO TO 98                                 SAMS0202
      KSUB=KN(JS,1)                                                     SAMS0203
      IF(MFAIL(KSUB,JP,JS) .EQ. 2         ) GO TO 99                    SAMS0204
      GO TO 98                                                          SAMS0205
   99 IFAIL(J+1,JS)=1                                                   SAMS0206
   98 CONTINUE                                                          SAMS0207
   90 CONTINUE                                                          SAMS0208
C
C          CHECKING FOR BUFFER FAILURE
C
      DO 4[illegible] JJ=1,[illegible]                                  SAMS0209
      IF(IBFAL(J-1,[illegible])-1) 5[illegible],4[illegible],59          SAMS0210
   59 CALL [illegible](I[illegible],I[illegible],[illegible])           SAMS0211
```

```
      IX=IY
      YY(JJ)=X
      IF(BPROB(JJ)-X) 41,40,40
   41 DO 42 L=1,N
      KSUB=KN(L,1)
      IF(MFAIL(KSUB,JJ,L)-1)43,44,43
   44 IFAIL(J+1,L)=1
      KSUB=KN(L, 1)
   43 IF(MFAIL(KSUB,JJ,L)-2) 42,46,42
   46 IFAIL(J+1,L)=1
   42 CONTINUE
      IBFAL(J,JJ)=1
   40 CONTINUE
      DO 105 IRS=1,NB
      IF(IBFAL(J-1,IRS)-1) 105,106,105
  106 CALL RANDU(IX,IY,X)
      IX=IY
      YY(IRS)=X
      IF(X-BPROB(IRS)) 105,105,107
  107 IBFAL(J,IRS)=1
      DO 52 L=1,N
      KSUB=KN(L,1)
      IF(MFAIL(KSUB,IRS,L) -1) 52,57,57
   57 IFAIL(J+1,L)=1
   52 CONTINUE
  105 CONTINUE
C
C         DETERMINATION OF UNIT SPEED CHANGE DUE TO BUFFER SIZE
C
      DO 110 I=1,NB
      RANGE=ULIM(I)-LOLIM(I)
      BLEV1=ULIM(I)-(SEN(I))*RANGE
      BLEV2=LOLIM(I)+(SEN(I))*RANGE
      IF(BUFF(J,I)-BLEV1) 112,111,111
  111 DO 120 KA=1,N
      KSUB=KN(KA,1)
      IF(MFAIL(KSUB,I,KA) .EQ. 0) GO TO 120
      IF(MFAIL(KSUB,I,KA)-2) 121,122,121
  122 KSLOW(KA)=1
      GO TO 120
  121 IF(MFAIL(KSUB,I,KA)-1) 120,123,120
  123 KFAST(KA)=1
  120 CONTINUE
  112 IF(BLEV2-BUFF(J,I)) 110,115,115
  115 DO 130 KB=1,N
      KSUB=KN(KB,1)
      IF(MFAIL(KSUB,I,KB) .EQ. 0) GO TO 130
      IF(MFAIL(KSUB,I,KB)-2) 131,132,131
  132 KFAST(KB)=1
      GO TO 130
  131 IF(MFAIL(KSUB,I,KB)-1) 130,133,130
  133 KSLOW(KB)=1
  130 CONTINUE
  110 CONTINUE
      KOT=KOT+1
      IF(KOT-NOUT) 62,145,62
  145 KOT=0
      KOUNT=KOUNT+1
```

```
      IF(KOUNT-2) 141,142,141                                           SAMS0267
  142 WRITE(IO,222)                                                     SAMS0268
  222 FORMAT('1',//)                                                    SAMS0269
      KOUNT=0                                                           SAMS0270
  141 IJK=J-1                                                           SAMS0271
C
C        PRINTING CURRENT PROCESS STATUS
C
      WRITE(IO,333) IJK                                                 SAMS0272
  333 FORMAT(//  ,20X,'TIME INTERVAL NO.',I3,/,20X, 20('-'),            SAMS0273
     1 /,01X,'     UNIT STATUS ',/,05X, 11('-'),          /,21X,'SHUT  SSAMS0274
     2PEED  SLOW  ENTER  EXIT   RANDOM '                                SAMS0275
     2                               ,/,' NUMBER  PRODUCTION  DOWN    UP  DOSAMS0276
     3WN  STATE  STATE  NUMBER '                                        SAMS0277
     3                       ,/,' ------  ----------  ----  -----  ----  ---SAMS0278
     4--  -----  ------  ')                                             SAMS0279
  444 FORMAT(1H ,1X,I2,6X,F7.1,5X,I1,6X,I1,5X,I1,5X,I2,5X,I2,4X,F5.3)   SAMS0280
      DO 140 LD=1,N                                                     SAMS0281
      WRITE(IO,444) LD,PROD(J,LD),IFAIL(J,LD),KFAST(LD),KSLOW(LD),      SAMS0282
     1 ISTAT(J-1,LD),ISTAT(J,LD),XX(LD)                                 SAMS0283
  140 CONTINUE                                                          SAMS0284
      WRITE(IO,555)                                                     SAMS0285
  555 FORMAT(/    ,02X,'BUFFER STATUS',/,02X,  13('-'),       /,' NUMBER  SAMS0286
     1  SIZE   FAIL  RANDOM NO. ',/,' ------  -------  ----  ---------- SAMS0287
     2 ')                                                               SAMS0288
      DO 150 LE=1,NB                                                    SAMS0289
      WRITE(IO,666) LE,BUFF(J,LE),IBFAL(J,LE),YY(LE)                    SAMS0290
  666 FORMAT(1H ,1X,I2,5X,F7.0,3X,I1,7X,F5.3)                           SAMS0291
  150 CONTINUE                                                          SAMS0292
   62 CONTINUE                                                          SAMS0293
C
C        PRODUCTION SUMMARY
C
  152 KOTT=KOTT+1                                                       SAMS0294
      IF(J-INTVL) 69,151,69                                             SAMS0295
   69 IF(KOTT-NPROD) 60,151,60                                          SAMS0296
  151 KOTT=0                                                            SAMS0297
  153 NTVL=J-1                                                          SAMS0298
      INT=NTVL                                                          SAMS0299
      DO 175 L=1,N                                                      SAMS0300
      SU=0.                                                             SAMS0301
      ZZ=0                                                              SAMS0302
      DO 185 LL=1,J                                                     SAMS0303
      IF(IFAIL(LL,L)-1) 185,183,184                                     SAMS0304
  183 ZZ=ZZ+1.                                                          SAMS0305
      GO TO 185                                                         SAMS0306
  184 SU=SU+1.                                                          SAMS0307
  185 CONTINUE                                                          SAMS0308
      TFAIL(L)=(ZZ/INT)*100.                                            SAMS0309
      SETUPS(L)=(SU/INT)*100.                                           SAMS0310
  175 CONTINUE                                                          SAMS0311
      DO 170 I=1,N                                                      SAMS0312
      Z=0                                                               SAMS0313
      Y=0                                                               SAMS0314
      DO 180 IA=2,J                                                     SAMS0315
      IF(PROD(IA,I)-0) 180,181,182                                      SAMS0316
  181 Z=Z+1.                                                            SAMS0317
  182 Y=Y+PROD(IA,I)                                                    SAMS0318
```

```
  160 CONTINUE                                                        SAMS0319
      IDOWN(I)=Z                                                      SAMS0320
      TPROD(I)=Y                                                      SAMS0321
      AVAM(I)=(1.-(TDOWN(I)/INT))*100.                                SAMS0322
  170 CONTINUE                                                        SAMS0323
      IF(KOTT-0) 172,171,172                                          SAMS0324
  171 WRITE(10,222)                                                   SAMS0325
      WRITE(10,766) NDAY                                              SAMS0326
  766 FORMAT(/////// ,25X,'DAY ',I3,/,25X,'-------')                  SAMS0327
      WRITE(10,767)                                                   SAMS0328
  767 FORMAT(         // ,20X,'PRODUCTION SUMMARY',/,20X,'---------------SAMS0329
     1---     ',//   ,10X,'   UNIT    TOTAL       AVAILIBILITY  PERCENT DOSAMS0330
     8WN    PERCENT DOWN'                                      ,/,10X,'SAMS0331
     2NUMBER   PRODUCTION     (PERCENT)     DUE TO BUFFERS  DUE TO SETUPS'SAMS0332
     )                                         ,/,10X,'-------  ----------SAMS0333
     3 -------------  -------------  -------------')                  SAMS0334
      DO 75 L=1,N                                                     SAMS0335
      WRITE(10,911) L,TPROD(L),AVAM(L),TFAIL(L),SETUPS(L)             SAMS0336
  911 FORMAT(1H ,10X,I2,07X,F8.0,08X,F4.0,10X,F4.0,10X,F4.0)          SAMS0337
   75 CONTINUE                                                        SAMS0338
  172 CONTINUE                                                        SAMS0339
C
C         FINDING NUMBER OF BUFFER FAILURES
C
      DO 270 I=1,NB                                                   SAMS0340
      XY=0                                                            SAMS0341
      DO 280 IA=2,J                                                   SAMS0342
      IF(IBFAL(IA,I)-1) 280,281,280                                   SAMS0343
  281 XY=XY+1.                                                        SAMS0344
  280 CONTINUE                                                        SAMS0345
      TBUFF(I)=XY                                                     SAMS0346
      AVAB(I)=100.*(1.-TBUFF(I)/INT)                                  SAMS0347
  270 CONTINUE                                                        SAMS0348
      IF(KOTT-0) 176,177,176                                          SAMS0349
  177 WRITE(10,583)                                                   SAMS0350
  583 FORMAT(///          ,10X,'BUFFER  INITIAL  FINAL    AVAILIBILITY',/,SAMS0351
     110X,'NUMBER   SIZE     SIZE      (PERCENT)',/,10X,'------  -------  SAMS0352
     2-------  ------------       ')                                  SAMS0353
      DO 290 I=1,JB                                                   SAMS0354
      WRITE(10,584) I,BUFF(1,I),BUFF(J,I),AVAB(I)                     SAMS0355
  584 FORMAT(1H ,10X,I2,5X,F7.0,2X,F7.0,5X,F4.0)                      SAMS0356
  290 CONTINUE                                                        SAMS0357
      IJ=J-1                                                          SAMS0358
      WRITE(10,587) IJ                                                SAMS0359
  587 FORMAT(// ,10X,'SUMMARY AFTER ',I3,' INTERVALS ',/,10X,'----------SAMS0360
     1-----------------   ')                                          SAMS0361
  176 CONTINUE                                                        SAMS0362
      KOUNT=1                                                         SAMS0363
   60 CONTINUE                                                        SAMS0364
      TOTPC(NDAY)=0                                                   SAMS0365
      IF(JSKIP-1) 999,61,999                                          SAMS0366
C
C         COST ANALYSIS
C
   61 CALL COST ([illegible],TDOWN,TBUFF,INTVL,TPROD ,NDAY,TOTPC(NDAY),BUFF)   SAMS0367
  999 IF(JPLOT-1) 919,[illegible],919                                 SAMS0368
  [illegible] KT=KT+1                                                 SAMS0369
      IF(JPLOT-KT) 919,[illegible],919                                SAMS0370
```

```
C
C        PLOTTING ROUTINE
C
  920 CALL XPLOT(INTVL,NB,BUFF,NPRNT,NDAY,ULIM)                    SAMS0371
      KT=0                                                          SAMS0372
C
C        STATISTICAL ANALYSIS
C
  919 DO 320 LK=1,N                                                 SAMS0373
      INT=INTVL-1                                                   SAMS0374
      XSUM=0                                                        SAMS0375
      DO 310 L=2,INTVL                                              SAMS0376
      XSUM=XSUM+PROD(L,LK)                                          SAMS0377
  310 CONTINUE                                                      SAMS0378
      XSUM=XSUM/INT                                                 SAMS0379
      AVGPR(NDAY,LK)=XSUM                                           SAMS0380
  320 CONTINUE                                                      SAMS0381
      DO 330 LK=1,N                                                 SAMS0382
      XXSUM=0                                                       SAMS0383
      DO 340 L=2,INTVL                                              SAMS0384
      XXSUM=XXSUM+((AVGPR(NDAY,LK)-PROD(L,LK))**2)                  SAMS0385
  340 CONTINUE                                                      SAMS0386
      XXSUM=XXSUM/INT                                               SAMS0387
      STDEV(NDAY,LK)=SQRT(XXSUM)                                    SAMS0388
  330 CONTINUE                                                      SAMS0389
      DO 350 L=1,N                                                  SAMS0390
      AVAMP(NDAY,L)=AVAM(L)                                         SAMS0391
  350 CONTINUE                                                      SAMS0392
      DO 360 LK=1,NB                                                SAMS0393
      YSUM=0                                                        SAMS0394
      DO 370 L=1,INTVL                                              SAMS0395
      YSUM=YSUM+BUFF(L,LK)                                          SAMS0396
  370 CONTINUE                                                      SAMS0397
      YSUM=YSUM/INTVL                                               SAMS0398
      AVGBU(NDAY,LK)=YSUM                                           SAMS0399
  360 CONTINUE                                                      SAMS0400
      DO 380 LK=1,NB                                                SAMS0401
      YYSUM=0                                                       SAMS0402
      DO 390 L=1,INTVL                                              SAMS0403
      YYSUM=YYSUM+((AVGBU(NDAY,LK)-BUFF(L,LK))**2)                  SAMS0404
  390 CONTINUE                                                      SAMS0405
      YYSUM=YYSUM/INTVL                                             SAMS0406
      STDEB(NDAY,LK)=SQRT(YYSUM)                                    SAMS0407
  380 CONTINUE                                                      SAMS0408
      DO 410 L=1,NB                                                 SAMS0409
      AVAMB(NDAY,L)=AVAB(L)                                         SAMS0410
  410 CONTINUE                                                      SAMS0411
      DO 420 L=1,N                                                  SAMS0412
      DOWN(NDAY,L)=TFAIL(L)                                         SAMS0413
  420 CONTINUE                                                      SAMS0414
C
 1000 CONTINUE                                                      SAMS0415
C
      DO 455 L=1,N                                                  SAMS0416
      XXSUM=0                                                       SAMS0417
      YYSUM=0                                                       SAMS0418
      ZZSUM=0                                                       SAMS0419
      TTSUM=0                                                       SAMS0420
```

```
      SSUM=0                                                            SAMS0421
      DO 465 K=1,NDAYS                                                  SAMS0422
      XXSUM=XXSUM+AVGPR(K,L)                                            SAMS0423
      YYSUM=YYSUM+AVANM(K,L)                                            SAMS0424
      ZZSUM=ZZSUM+((STDEM(K,L))**2)                                     SAMS0425
      TTSUM=TTSUM+DOWN(K,L)                                             SAMS0426
      SSUM=SSUM+TOTPC(K)                                                SAMS0427
  465 CONTINUE                                                          SAMS0428
      TAVG(L)=XXSUM/NDAYS                                               SAMS0429
      TAVAM(L)=YYSUM/NDAYS                                              SAMS0430
      TSTDM(L)=SQRT(ZZSUM)                                              SAMS0431
      TTDOWN(L)=TTSUM/NDAYS                                             SAMS0432
      TTPC=SSUM/NDAYS                                                   SAMS0433
  455 CONTINUE                                                          SAMS0434
      DO 475 L=1,NB                                                     SAMS0435
      XXSUM=0                                                           SAMS0436
      YYSUM=0                                                           SAMS0437
      ZZSUM=0                                                           SAMS0438
      DO 485 K=1,NDAYS                                                  SAMS0439
      XXSUM=XXSUM+AVGOU(K,L)                                            SAMS0440
      YYSUM=YYSUM+((STDEB(K,L))**2)                                     SAMS0441
      ZZSUM=ZZSUM+AVAMB(K,L)                                            SAMS0442
  485 CONTINUE                                                          SAMS0443
      TAVGB(L)=XXSUM/NDAYS                                              SAMS0444
      TSTDB(L)=SQRT(YYSUM)                                              SAMS0445
      TAVAB(L)=ZZSUM/NDAYS                                              SAMS0446
  475 CONTINUE                                                          SAMS0447
C
C         PRINTING PROCESS STATICS
C
      WRITE(IO,591) (NO(K),K=1,N)                                       SAMS0448
  591 FORMAT('1',////// , 9X,'AVERAGE   UNIT  PRODUCTION/INTERVAL ',/,9SAMS0449
     3X,'-----------------------------------  '                         SAMS0450
     1                           ,/,15X,'DAY  COST/UNIT($)',01X,10(3X,I2,2XSAMS0451
     2))                                                                SAMS0452
      WRITE(IO,453) (NX(K),K=1,N)                                       SAMS0453
  453 FORMAT(15X,'---  ------------',10(2X,A1,'----'))                  SAMS0454
      DO 450 K=1,NDAYS                                                  SAMS0455
      WRITE(IO,451) K,TOTPC(K),(AVGPR(K,L),L=1,N)                       SAMS0456
  451 FORMAT(11X,I2,06X,F5.2,3X,10(2X,F5.0))                            SAMS0457
  450 CONTINUE                                                          SAMS0458
      WRITE(IO,454) (NX(K),K=1,N)                                       SAMS0459
  454 FORMAT(15X,'------------',10(2X,A1,'----'))                       SAMS0460
      WRITE(IO,452) TTPC,(TAVG(L),L=1,N)                                SAMS0461
  452 FORMAT(17X,F5.2,05X,10(F5.0,2X))                                  SAMS0462
      WRITE(IO,592)                                                     SAMS0463
  592 FORMAT('1',////// ,15X,'PRODUCTION DEVIATION ',/,15X,'-----------SAMS0464
     1---------  ')                                                     SAMS0465
      WRITE(IO,599) (NO(K),K=1,N)                                       SAMS0466
  599 FORMAT(15X,'DAY',02X,10(2X,I2,3X))                                SAMS0467
      WRITE(IO,456) (NX(K),K=1,N)                                       SAMS0468
  456 FORMAT(15X,'---',10(2X,A1,'----'))                                SAMS0469
      DO 430 K=1,NDAYS                                                  SAMS0470
      WRITE(IO,431) K,(STDEM(K,L),L=1,N)                                SAMS0471
  431 FORMAT(11X,I2,2X,10(F5.0,2X))                                     SAMS0472
  430 CONTINUE                                                          SAMS0473
      WRITE(IO,457) (NX(K),K=1,N)                                       SAMS0474
  457 FORMAT(13X,10(2X,A1,'----'))                                      SAMS0475
```

```
      WRITE(10,432) (TSTDM(L),L=1,N)                                    SAMS0476
  432 FORMAT(15X,10(F5.1,2X))                                           SAMS0477
      WRITE(10,593)                                                     SAMS0478
  593 FORMAT('1',//////   ,15X,'PRODUCTION UNIT AVAILABILITY  ',/,15X,  SAMS0479
     1'----------------------------   ')                                SAMS0480
      WRITE(10,599) (NU(K),K=1,N)                                       SAMS0481
      WRITE(10,456) (NX(K),K=1,N)                                       SAMS0482
      DO 440 K=1,NDAYS                                                  SAMS0483
      WRITE(10,431) K,(AVAMM(K,L),L=1,N)                                SAMS0484
  440 CONTINUE                                                          SAMS0485
      WRITE(10,457) (NX(K),K=1,N)                                       SAMS0486
      WRITE(10,432) (TAVAM(L),L=1,N)                                    SAMS0487
      WRITE(10,597)                                                     SAMS0488
  597 FORMAT('1',////// ,13X,'   UNIT DOWNTIME DUE TO BUFFERS  ',/,13X,SAMS0489
     1'    ----------------------------   ')                            SAMS0490
      WRITE(10,599) (NU(K),K=1,N)                                       SAMS0491
      WRITE(10,456) (NX(K),K=1,N)                                       SAMS0492
      DO 460 K=1,NDAYS                                                  SAMS0493
      WRITE(10,431) K,(DOWN(K,L),L=1,N)                                 SAMS0494
  460 CONTINUE                                                          SAMS0495
      WRITE(10,457) (NX(K),K=1,N)                                       SAMS0496
      WRITE(10,432) (TDOWN(L),L=1,N)                                    SAMS0497
      WRITE(10,594)                                                     SAMS0498
  594 FORMAT('1',//////  ,15X,'AVERAGE BUFFER SIZE  ',/,15X,'-----------SAMS0499
     1--------   ')                                                     SAMS0500
      WRITE(10,599) (NU(K),K=1,NB)                                      SAMS0501
      WRITE(10,456) (NX(K),K=1,NB)                                      SAMS0502
      DO 470 K=1,NDAYS                                                  SAMS0503
      WRITE(10,471) K,(AVGBU(K,L),L=1,NB)                               SAMS0504
  471 FORMAT(11X,I2,10(1X,F6.0))                                        SAMS0505
  470 CONTINUE                                                          SAMS0506
      WRITE(10,457) (NX(K),K=1,NB)                                      SAMS0507
      WRITE(10,472) (TAVGB(L),L=1,NB)                                   SAMS0508
  472 FORMAT(14X,10(F6.0,1X))                                           SAMS0509
      WRITE(10,595)                                                     SAMS0510
  595 FORMAT('1',//////   ,15X,'BUFFER DEVIATION  ',/,15X,'-------------SAMS0511
     1---   ')                                                          SAMS0512
      WRITE(10,599) (NU(K),K=1,NB)                                      SAMS0513
      WRITE(10,456) (NX(K),K=1,NB)                                      SAMS0514
      DO 480 K=1,NDAYS                                                  SAMS0515
      WRITE(10,471) K,(STDEB(K,L),L=1,NB)                               SAMS0516
  480 CONTINUE                                                          SAMS0517
      WRITE(10,457) (NX(K),K=1,NB)                                      SAMS0518
      WRITE(10,472) (TSTDB(L),L=1,NB)                                   SAMS0519
      WRITE(10,596)                                                     SAMS0520
  596 FORMAT('1',////// ,15X,'BUFFER AVAILABILITY   ',/,15X,'-----------SAMS0521
     1--------   ')                                                     SAMS0522
      WRITE(10,599) (NU(K),K=1,NB)                                      SAMS0523
      WRITE(10,456) (NX(K),K=1, NB)                                     SAMS0524
      DO 490 K=1, NDAYS                                                 SAMS0525
      WRITE(10,431) K,(AVAMB(K,L),L=1,NB)                               SAMS0526
  490 CONTINUE                                                          SAMS0527
      WRITE(10,457) (NX(K),K=1,NB)                                      SAMS0528
      WRITE(10,432) (TAVAB(L),L=1,NB)                                   SAMS0529
      STOP                                                              SAMS0530
      END                                                               SAMS0531
      SUBROUTINE [illegible](J, SETUP,IFAIL,K)                          SAMS0532
      DIMENSION SET[illegible](10,5),K(10,3),IFAIL(481,10)              SAMS0533
```

```
      COMMON IN,IO,IP,KN,NJ                                              SAMS0534
      KSUB=KN(K,2)                                                       SAMS0535
      MSUB=IFIX(SETUP(K,KSUB))                                           SAMS0536
      DO 10 KK=1,MSUB                                                    SAMS0537
      JSUB=J+KK                                                          SAMS0538
   10 IFAIL(JSUB,K)=2                                                    SAMS0539
      IDUM=KN(K,1)                                                       SAMS0540
      JIMES=NJ-1                                                         SAMS0541
      DO 20 JZ=1,JIMES                                                   SAMS0542
   20 KN(K,JZ)=KN(K,JZ+1)                                                SAMS0543
      KN(K,NJ)=IDUM                                                      SAMS0544
      RETURN                                                             SAMS0545
      END                                                                SAMS0546
C MARKOV MATRIX CALCULATION
C
      SUBROUTINE MKVBLD(SMNUM,B,POUT,SMTYPE,LIMIT,IX)                    SAMS0547
      DIMENSION    B(21,23,10),POUT(10,20),KN(10,3),ADJUS(10),NAME(13)   SAMS0548
      DIMENSION RATS(10),IATS(2,10),LIMIT(10,3)                          SAMS0549
      INTEGER CLCMKV,RDMKV,PCHMKV,PRTMKV,SMNUM,SMTYPE,X,F,Y,Z            SAMS0550
      DATA MA 71HA7,MB 71HB7                                             SAMS0551
      COMMON IN,IO,IP,KN,NJ,ADJUS,NRBMAT,NCBMAT,RATS,IATS                SAMS0552
C
C SMNUM  SIMULATION PRODUCTION UNIT NUMBER
C SMTYPE  SIMULATION PRODUCTION UNIT TYPE
C CLCMKV  CALCULATE MARKOV MATRIX
C RDMKV  READ MARKOV MATRIX
C PCHMKV  PUNCH MARKOV MATRIX
C PRTMKV  PRINT MARKOV MATRIX
C NAME  IDENTIFICATION OF SIMULATION PRODUCTION UNIT
C IX  SEED FOR RANDOM PRODUCTION GENERATION
C B MATRIX  CALCULATION OF TRANSFER PROBABILITIES FROM STATE TO STATE FOR EACH
C       PRODUCTION UNIT. MAXIMUM OF 20 TOTAL STATES IN EACH OF 10 PRODUCTION UNITS
C POUT  PRODUCTION LEVEL AT DIFFERENT STATES
C KN      ORDER OF JOBS TO BE PRODUCED
C
      READ (IN,1000) CLCMKV,RDMKV,PCHMKV,PRTMKV,SMNUM,SMTYPE,NAME,IX     SAMS0553
     *,(KN(SMNUM,L),L=1,NJ).(LIMIT(SMNUM,L),L=1,NJ)                     SAMS0554
 1000 FORMAT(4I2,2X,2I2,1X,13A2,4X,I5,5X,3I2,4X,3F5.0)                   SAMS0555
      IF(CLCMKV .EQ. 0) GO TO 80                                         SAMS0556
C
C CALCULATE MARKOV MATRIX
C
       GO TO (10,20,30,40),SMTYPE                                        SAMS0557
C
C SIMULATION MODEL A
C N PARALLEL PRODUCTION LINES WITH K STATION ON EACH LINE.
C SIZE OF MATRIX IS DEPENDENT ON THE REPAIR POLICY USED  (X+2) STATES
C
C  N  NUMBER OF PARALLEL PRODUCTION LINES
C  X  NUMBER OF MINOR FAILURES OCCURING BEFORE SHUT DOWN FOR MINOR REPAIRS
C  R  PROBABILITY (X MINOR REPAIRS IN A TIME INTERVAL)
C  P  PROBABILITY (A MINOR FAILURE IN A TIME INTERVAL)
C PM  PROBABILITY (A MAJOR FAILURE IN A TIME INTERVAL)
C RM  PROBABILITY (A MAJOR REPAIR  IN A TIME INTERVAL)
C ROP RATE OF PRODUCTION
C
   10 READ (IN,100) N,X,P,R,PM,PM,ROP                                    SAMS0558
  100 FORMAT (2I2,6X,5F10.0)                                             SAMS0559
```

```
      L=1                                                              SAMS0560
C CALCULATION FO TRANSITION PROBABILITIES FOR AREA I
      DO 11 I=1,X                                                      SAMS0561
      DO 12 J=L,X                                                      SAMS0562
      B(SMNUM,I,J)=(1.-PM)*((FAC(N-I+1,J-I))*(P**(J-I))*(1.-P)**(N-J+1))SAMS0563
   12 CONTINUE                                                         SAMS0564
      J=X+1                                                            SAMS0565
      KL=(X-I+1)+1                                                     SAMS0566
      KU=(N-I+1)+1                                                     SAMS0567
      SUM=0.                                                           SAMS0568
C CALCULATION OF TRANSITION PROBABILITIES FOR AREA II
      DO 13 KI=KL,KU                                                   SAMS0569
      K=KI                                                             SAMS0570
      SUM=SUM+FAC(N-J+1,K)*P**K*(1.-P)**(N-I+K+1)                      SAMS0571
   13 CONTINUE                                                         SAMS0572
      B(J,I,SMNUM)=(1.-PM)*(SUM*R**K*(1.-R)**(N-J+K+1))                SAMS0573
      J=X+2                                                            SAMS0574
      B(J,I,SMNUM)=PM                                                  SAMS0575
      L=L+1                                                            SAMS0576
   11 CONTINUE                                                         SAMS0577
      B(X+1,1,SMNUM)=R                                                 SAMS0578
      B(X+2,1,SMNUM)=RM                                                SAMS0579
      B(X+1,X+1,SMNUM)=1.-R                                            SAMS0580
      B(X+2,X+2,SMNUM)=1.-RM                                           SAMS0581
      GO TO 40                                                         SAMS0582
C
C  SIMULATION MODEL B
C N PARALLEL PRODUCTION LINES WITH K STATIONS ON EACH LINE (WHERE SOME STATIONS
C     ARE SUBDIVIDED INTO M PARALLEL LINES)
C SIZE OF MATRIX IS DEPENDENT UPON REPAIR POLICY USED  NM+2 STATES
C
C  N   NUMBER OF PARALLEL PRODUCTION LINES
C  F   NUMBER OF STATION FAILURES OCCURING BEFORE SHUTDOWN FOR MINOR AND STATION
C      REPAIRS
C PS   PROBABILITY (A SUBLINE FAILURE IN A TIME INTERVAL)
C RS   PROBABILITY (F SUBLINE REPAIRS IN A TIME INTERVAL)
C  M   NUMBER OF SUBLINES THAT CERTAIN STATIONS ARE DIVIDED INTO
C
   20 READ (IN,200) N,F,M,P,R,PM,RM,PS,RS,ROP                          SAMS0583
  200 FORMAT(3I2,4X,7F10.0)                                            SAMS0584
      L=1                                                              SAMS0585
      DO 21 I=1,F                                                      SAMS0586
      DO 22 J=L,F                                                      SAMS0587
      KU=I/M +1                                                        SAMS0588
      SUM1=0.                                                          SAMS0589
      SUM2=0.                                                          SAMS0590
      KYU=((J-I)/M)+1                                                  SAMS0591
      DO 23 KI=1,KU                                                    SAMS0592
      K=KI-1                                                           SAMS0593
      VALUE= FAC(I-(M-1)*K,K)*(P**K)*(PS**(I-M*K))                     SAMS0594
      SUM1=SUM1+VALUE                                                  SAMS0595
      SUM3=0.                                                          SAMS0596
      DO 24 KY=1,KYU                                                   SAMS0597
      Y=KY-1                                                           SAMS0598
      SUM3=SUM3 + FAC(N-K,Y)*(P**Y)*((1.-P)**(N-K-Y)) * FAC(K*N-I+M*K, SAMS0599
     *J-I-M*Y)*(PS**(J-I-M*Y))*((1.-PS)**(K*N-J+M*K+M*Y))              SAMS0600
   24 CONTINUE                                                         SAMS0601
```

```
      SUM2=SUM2+(VALUE*SUM3)                                            SAMS0602
   23 CONTINUE                                                          SAMS0603
      B(I,J,SMNUM) = (1.-PM) * (1.0 / SUM1) + SUM2                      SAMS0604
   22 CONTINUE                                                          SAMS0605
      J= F+1                                                            SAMS0606
      SUM1=0.                                                           SAMS0607
      SUM2=0.                                                           SAMS0608
      DO 25 KI=1,KU                                                     SAMS0609
      K= KI-1                                                           SAMS0610
      SUM1=SUM1 + FAC(I-(M-1)*K,K)*(P**K)*((1-P)**(I-M*K))              SAMS0611
      VAL1 = FAC(I-(M-1)*K,K)* (P**K)* (PS**(I-M*K))                    SAMS0612
      KYU = N-K+1                                                       SAMS0613
      SUM3=0.                                                           SAMS0614
      DO 26 KY=1,KYU                                                    SAMS0615
      Y=KY-1                                                            SAMS0616
      VAL2=FAC(N-K,Y)*(P**Y)*(1-P)                                      SAMS0617
      KYZU=N*K-I+M*K                                                    SAMS0618
      KYZL=N-I-M*Y                                                      SAMS0619
      SUM4=0.                                                           SAMS0620
      DO 27 Z=KYZL,KYZU                                                 SAMS0621
      VAL4 = FAC(N*K-I+M*K,Z) * (PS**Z) * ((1-PS)**(N*K-I+M*K-Z))       SAMS0622
      SUM4 = SUM4 + VAL4                                                SAMS0623
   27 CONTINUE                                                          SAMS0624
      SUM3 = SUM3 + (VAL2*SUM4)                                         SAMS0625
   26 CONTINUE                                                          SAMS0626
      SUM2 = SUM2 + (VAL1*SUM3)                                         SAMS0627
   25 CONTINUE                                                          SAMS0628
      B(I,J,SMNUM) = (1.-PM) * (1./SUM1) * SUM2                         SAMS0629
      J = F+2                                                           SAMS0630
      B(I,J,SMNUM) = PM                                                 SAMS0631
   21 CONTINUE                                                          SAMS0632
      B(F+1,1,SMNUM)=RS                                                 SAMS0633
      B(F+2,1,SMNUM)=RM                                                 SAMS0634
      B(F+1,F+1,SMNUM)=(1.-RS)                                          SAMS0635
      B(F+2,F+2,SMNUM)=1.-RM                                            SAMS0636
      GO TO 86                                                          SAMS0637
C
C SIMULATION MODEL C
C N PARALLEL PRODUCTION LINES WITH K STATIONS ON EACH LINE (NO STATIONS  ARE
C      SUBDIVIDED)
C REPAIRS CAN BE EFFECTED WHILE THE PRODUCTION UNIT IS IN OPERATION
C SIZE OF MATRIX IS FIXED UPON THE NUMBER OF PRODUCTION LINES  N+2 STATES
C  N  NUMBER OF PARALLEL PRODUCTION LINES
C  P  PROBABILITY (A MINOR FAILURE IN A TIME INTERVAL)
C  R  PROBABILITY (A MINOR REPAIR  IN A TIME INTERVAL)
C PM  PROBABILITY (A MAJOR FAILURE IN A TIME INTERVAL)
C RM  PROBABILITY (A MAJOR REPAIR  IN A TIME INTERVAL)
C
   30 READ(IN,300) N,P,R,PM,RM,ROP                                      SAMS0638
  300 FORMAT(I2,8X,5F10.0)                                              SAMS0639
      L=1                                                               SAMS0640
      NI=N+1                                                            SAMS0641
      DO 31 I=1,NI                                                      SAMS0642
      DO 32 J=L,NI                                                      SAMS0643
      SUM=0.                                                            SAMS0644
      KU=I-I+1                                                          SAMS0645
      DO 33 KI=1,KU                                                     SAMS0646
      K=KI-1                                                            SAMS0647
```

```
      VAL1 = FAC(N-I+1,K+J-1)*(P**(K+J-1))* ((1.-P)**(N-K-J+1))          SAMS0648
      VAL2 = FAC(I-1,K) * (R**K)*((1.-R)**(I-K-1))                       SAMS0649
      SUM = SUM + VAL1*VAL2                                              SAMS0650
   33 CONTINUE                                                           SAMS0651
      B(I,J,SMNUM) = (1.-PM)*SUM                                         SAMS0652
   32 CONTINUE                                                           SAMS0653
      B(I,N+2,SMNUM)=PM                                                  SAMS0654
      IF (I .EQ. 1) GO TO 66                                             SAMS0655
      DO 34 J=1,L2                                                       SAMS0656
      SUM = 0.                                                           SAMS0657
      KU=J-I+1                                                           SAMS0658
      DO 35 K1=1,KU                                                      SAMS0659
      K= K1-1                                                            SAMS0660
      VAL1 = (FAC(N-J+1,K)*(P**K)*((1.-P)**(N-I+1-K)))*(FAC(I-1,I-J+K) *SAMS0661
     *R**(I-J+K)*((1.-R)**(J-1-K)))                                      SAMS0662
      SUM= SUM + VAL1                                                    SAMS0663
   35 CONTINUE                                                           SAMS0664
      B(I,J,SMNUM) = (1.-PM)*SUM                                         SAMS0665
   34 CONTINUE                                                           SAMS0666
   66 CONTINUE                                                           SAMS0667
      L2=L                                                               SAMS0668
      L= L+1                                                             SAMS0669
   31 CONTINUE                                                           SAMS0670
      B(N+2,1,SMNUM)=RM                                                  SAMS0671
      B(N+2,N+2,SMNUM) = 1.-RM                                           SAMS0672
      GO TO 80                                                           SAMS0673
   40 CONTINUE                                                           SAMS0674
C SIMULATION MODEL SAMS
      CALL SAMBLD (SMNUM    ,B    ,NRBMAT,NCBMAT,RATS,IATS)              SAMS0675
   80 CONTINUE                                                           SAMS0676
      IF (RDMKV .EQ. 0) GO TO 90                                         SAMS0677
C READING IN A MARKOV MATRIX
      GO TO (81,82,83,84),SMTYPE                                         SAMS0678
   81 READ (IN,100) N,X,P,R,PM,RM,ROP                                    SAMS0679
      NROW= X+2                                                          SAMS0680
      NCOL=NROW                                                          SAMS0681
      GO TO 88                                                           SAMS0682
   82 READ (IN,200) N,F,M,P,R,PM,RM,PS,RS,ROP                            SAMS0683
      NROW= F+2                                                          SAMS0684
      NCOL=NROW                                                          SAMS0685
      GO TO 88                                                           SAMS0686
   83 READ(IN,300) N,P,R,PM,RM,ROP                                       SAMS0687
      NROW= N+2                                                          SAMS0688
      NCOL=NROW                                                          SAMS0689
      GO TO 88                                                           SAMS0690
   84 NROW=21                                                            SAMS0691
      NCOL=23                                                            SAMS0692
   88 CONTINUE                                                           SAMS0693
      DO 87 I=1,NROW                                                     SAMS0694
   87 READ (IN,101) ( B(I,J,SMNUM),J=1,NCOL)                             SAMS0695
  101 FORMAT (11F7.4)                                                    SAMS0696
   90 CONTINUE                                                           SAMS0697
      IF (PCHMKV .EQ. 0) GO TO 99                                        SAMS0698
C
C PUNCH OUT MARKOV MATRIX
C
      GO TO (91,92,93,94),SMTYPE                                         SAMS0699
   91 NROW= X+2                                                          SAMS0700
```

```
      NCOL=NROW                                                         SAMS0701
      GO TO 98                                                          SAMS0702
   92 NROW= F+2                                                         SAMS0703
      NCOL=NROW                                                         SAMS0704
      GO TO 98                                                          SAMS0705
   93 NROW= N+2                                                         SAMS0706
      NCOL=NROW                                                         SAMS0707
      GO TO 98                                                          SAMS0708
   94 NROWS=21                                                          SAMS0709
      NCOLS=23                                                          SAMS0710
   98 CONTINUE                                                          SAMS0711
      DO 97 I=1,NROW                                                    SAMS0712
   97 WRITE(IP,101) (B(I,J,SMNUM),J=1,NCOL)                             SAMS0713
   99 CONTINUE                                                          SAMS0714
C ..
C CALCULATING PRODUCTION FOR EACH STATE OF THE DIFFERENT MODELS
C      EACH LINE PRODUCES ROP UNITS PER TIME INTERVAL
C
      GO TO (71,72,73,73),SMTYPE                                        SAMS0715
C
C PRODUCTION FOR MODEL A   XMINOR FAILURES ALLOWED
C
   71 DO 75 I=1,X                                                       SAMS0716
   75 POUT(SMNUM,I) = (N-I+1)*ROP                                       SAMS0717
      GO TO 79                                                          SAMS0718
C
C PRODUCTION FOR MODEL B   F STATION FAILURES OR F/M MINOR FAILURES ALLOWED
C
   72 DO 76 I=1,F                                                       SAMS0719
   76 POUT(SMNUM,I)=(N*L)-((I-1)/M)*ROP                                 SAMS0720
      GO TO 79                                                          SAMS0721
C
C PRODUCTION FOR MODEL C
C
   73 DO 77 I=1,N                                                       SAMS0722
   77 POUT(SMNUM,I)=(N-I+1)*ROP                                         SAMS0723
   79 CONTINUE                                                          SAMS0724
      IF (PRTMRKV .EQ. 0) GO TO 599                                     SAMS0725
C
C PRINT OUT THE MARKOV MATRIX
C
      WRITE (IO,510) SMNUM,NAME                                         SAMS0726
  510 FORMAT (1H1,16HPRODUCTION UNIT ,I2,//,3X,13A2)                    SAMS0727
      IF (SMTYPE - 2) 41,42,43                                          SAMS0728
C
C MODEL A PRINT
C
   41 WRITE (IO,541) NA,N,X,P,R,PM,RM,ROP                               SAMS0729
  541 FORMAT(//,6X,16HPRODUCTION MODEL,1X,A2,//,6X,I9,1X,               SAMS0730
     *15HPRODUCTION LINES                                               SAMS0731
     *   ,/,6X,I9,1X,33HFAILURES ALLOWED BEFORE SHUT DOWN,/,6X,F9.5,1X, SAMS0732
     *30HPROBABILITY OF A MINOR FAILURE,/,6X,F9.5,1X,28HPROBABILITY OF MSAMS0733
     *INOR REPAIR,/,6X,F9.5,1X,30HPROBABILITY OF A MAJOR FAILURE,/,6X, SAMS0734
     *F9.5,1X,29HPROBABILITY OF A MAJOR REPAIR,/,6X,F9.5,1X,27HRATE OF PSAMS0735
     *RODUCTION PER LINE  )                                             SAMS0736
      IX= N+2                                                           SAMS0737
      WRITE(IO,500)                                                     SAMS0738
  500  FORMAT(1H ,13HMARKOV MATRIX,//)                                  SAMS0739
```

```
      DO 44 I=1,IX                                                      SAMS0740
   44 WRITE (10,501) (B(I,J,SMNUM),J=1,IX)                              SAMS0741
  501 FORMAT(1H ,20F7.4)                                                SAMS0742
      GO TO 599                                                         SAMS0743
C
C MODEL B PRINT
C
   42 WRITE (10,541) MB,N,F,P,R,PM,RM ,ROP                              SAMS0744
      WRITE(10,542) M,PS,RS                                             SAMS0745
  542 FORMAT(/,6X,I5,1X,45HNUMBER OF SUBLINES THAT CERTAIN STATIONS HAVESAMS0746
     *,/,6X,F5.4,1X,30HPROBABILITY OF STATION FAILURE,/,6X,F5.4,1X,  30HSAMS0747
     *PROBABILITY OF STATION REPAIR  )                                  SAMS0748
      WRITE(10,500)                                                     SAMS0749
      IF= F+2                                                           SAMS0750
      DO 45 I=1,IF                                                      SAMS0751
   45 WRITE (10,501) (B(I,J,SMNUM),J=1,IX)                              SAMS0752
      GO TO 599                                                         SAMS0753
C
C MODEL C PRINT
C
   43 WRITE (10,543) N,P,R,PM,RM,ROP                                    SAMS0754
  543 FORMAT(//,6X,18HPRODUCTION MODEL C,//,6X,I5,1X,16HPRODUCTION LINESSAMS0755
     *,/,6X,F5.4,1X,30HPROBABILITY OF A MINOR FAILURE,/,6X,F5.4,1X,  28HSAMS0756
     *PROBABILITY OF MINOR REPAIRS,/,6X,F5.4,1X,30HPROBABILITY OF A MAJOSAMS0757
     *R FAILURE,/,6X,F9.5,1X,29HPROBABILITY OF A MAJOR REPAIR,/,6X,F9.5,SAMS0758
     *1X,27HRATE OF PRODUCTION PER LINE  )                              SAMS0759
      WRITE(10,500)                                                     SAMS0760
      NI=N+1                                                            SAMS0761
      DO 46 I=1,NI                                                      SAMS0762
   46 WRITE (10,501) (B(I,J,SMNUM),J=1,IX)                              SAMS0763
  599 CONTINUE                                                          SAMS0764
C
C SETTING UP THE CUMULATIVE DISTRIBUTION
C
      GO TO(111,112,113,119),SMTYPE                                     SAMS0765
  111 NROWS=X+2                                                         SAMS0766
  116 CONTINUE                                                          SAMS0767
      NCOLS=NROWS                                                       SAMS0768
      DO 114 I=1,NROWS                                                  SAMS0769
      CUM=0.                                                            SAMS0770
      DO 115 J=1,NCOLS                                                  SAMS0771
      B(I,J,SMNUM)=B(I,J,SMNUM) + CUM                                   SAMS0772
      CUM = B(I,J,SMNUM)                                                SAMS0773
  115 CONTINUE                                                          SAMS0774
  114 CONTINUE                                                          SAMS0775
      GO TO 119                                                         SAMS0776
  112 NROWS= F+2                                                        SAMS0777
      GO TO 116                                                         SAMS0778
  113 NROWS=N+1                                                         SAMS0779
      GO TO 116                                                         SAMS0780
  119 CONTINUE                                                          SAMS0781
      RETURN                                                            SAMS0782
      END                                                               SAMS0783
C
C  SIMULATION OF PRODUCTION
C
      SUBROUTINE SMPROD (I1   ,B,XX,ISTATN,IFAIL,PROD,ISTATL,I2   ,     SAMS0784
     * POUT,KFAST,KSLOW,IY)                                             SAMS0785
```

```
      DIMENSION    B(10,20,20),POUT(10,20),ADJUS(10)
      INTEGER SMNUM,SMTYPE
      COMMON ADJUS
      SMNUM=I1
      SMTYPE=I2
C
      IF (IFAIL - 1) 5,3,3
    3 PROD= 0.
      GO TO (4,6),IFAIL
    4 CALL RANDU (IX,IY,X)
      IX=IY
      XX=X
      DO 7 J=1,20
      IF ( X .GT. B(SMNUM,ISTATL,J)) GO TO 7
      ISTATN = J
      GO TO 9
    7 CONTINUE
      GO TO 9
    6 ISTATN = 1
      GO TO 9
    5 CALL RANDU (IX,IY,X)
      IX= IY
      XX=X
      DO 8 J=1,20
      IF ( X .GT. B(SMNUM,ISTATL,J)) GO TO 8
      ISTATN = J
      IF ( KFAST         .EQ. 1 ) GO TO 15
      IF ( KSLOW .EQ. 1 ) GO TO 16
      PROD = POUT(SMNUM,J)
      GO TO 9
   15 PROD= POUT(SMNUM,J)+(ADJUS(SMNUM)*POUT(SMNUM,J))
      GO TO 9
   16 PROD= POUT(SMNUM,J)-(ADJUS(SMNUM)*POUT(SMNUM,J))
      GO TO 9
    8 CONTINUE
    9 CONTINUE
      RETURN
      END
      FUNCTION FAC (I,J)
      ISUM= 1
      JSUM= 1
      IF ( I .EQ. J ) GO TO 20
      IF ( J .EQ. 0 ) GO TO 20
      JL= I-J
      DO 10 K= 1,JL
   10 JSUM= JSUM * K
   15 CONTINUE
      IF ( I .EQ. 0 ) GO TO 20
      ILOW=    J+1
      DO 16 K=ILOW,I
   16 ISUM= ISUM * K
   20 FAC= ISUM/JSUM
      RETURN
      END
      SUBROUTINE MKMAT
      DIMENSION RHLD(21)
      DIMENSION A(1)
      INTEGER ITTS(3)
```

```
      DIMENSION B(1), NAME(10), RTTS(7)
      REAL RV(1)
      INTEGER IV(1)
      INTEGER CPNCH
      INTEGER SMNUM
      INTEGER MKVIO(4),CLCMKV,RDMKV,PCHMKV,PRTMKV
      EQUIVALENCE (MKVIO(1), CLCMKV), (MKVIO(2), RDMKV),
     +            (MKVIO(3), PCHMKV),(MKVIO(4),PRTMKV)
      EQUIVALENCE (ITTS(1),NRTSTA),(ITTS(2),NPOL),(ITTS(3),ISEED)
      EQUIVALENCE (RTTS(1),RMTBF),(RTTS(2),RMTTR),(RTTS(3),SMTBF),
     +            (RTTS(4),SMTTR),(RTTS(5),  TSS),(RTTS(6),   TJ),
     +            (RTTS(7),  RPM)
      EQUIVALENCE (ISEED,IX), (NPOL,NSHD)
      EQUIVALENCE (IN,IRDR),(IO,IPTR)
      COMMON IN,IO
      DATA CPNCH / 7 /
      IJLOC(IDUM,JDUM) = NRTNC*(SMNUM-1) + NR*(JDUM-1) + IDUM
      PROB(RMT)=1.0 - 1.0 / EXP(1.0/RMT)
      ENTRY SAMBLD (SMNUM,A,NR,NC,RV,IV)
      READ (IRDR,1000) MKVIO,SMNUM,NAME
      READ(IRDR,1010) ITTS,RTTS
      NRTNC = NR*NC
      NSTS = 2*NPOL + 1
      ICPRO = NSTS +1
      ICCUM = NSTS+2
      RV(SMNUM) = RMTTR
      I=2*SMNUM
      IV(I) = ISEED
      IV(I-1) = NPOL
      P   = PROB(RMTBF)
      R   = PROB(RMTTR)
      PS = PROB(SMTBF)
      RS = PROB(SMTTR)
      IF (RDMKV .EQ. 1) GO TO 80
C      ZERO OUT NECESSARY AREA OF MARKOV MATRIX
      LOC=IJLOC(1,1)
      DO 10 I=1,ICCUM
      LOC2 = (I-1)*NR + LOC
      LOC3 = LOC2 + NSTS
      DO 11 J=LOC2,LOC3
      A(J) = 0.0
   11 CONTINUE
   10 CONTINUE
      DO 40 I=1,NPOL
      LOC = IJLOC(I,I)
      A(LOC)=(1.0-P)**(NRTSTA-(I-1))
      SUM = A(LOC)
      NTF = 0
      NPM1 = NPOL - 1
      IF (I - NPM1) 15,15,25
   15 CONTINUE
      LOC1 = LOC
      DO 20 J=1,NPM1
      LOC = LOC1 + NR
      A(LOC)=A(LOC1)*((P*(NRTSTA-(I-1)-NTF))/((NTF+1)*(1.0-P)))
      SUM = SUM + A(LOC )
      LOC1 = LOC
      NTF = NTF + 1
```

```
   20 CONTINUE                                                        SAMS0900
   25 CONTINUE                                                        SAMS0901
      NP1 = NPOL + 1                                                  SAMS0902
      LOC = LOC  + NR                                                 SAMS0903
      A(LOC) = 1. - SUM                                               SAMS0904
      LOC = IJLOC(1,1)                                                SAMS0905
      DO 30 J=1,NP1                                                   SAMS0906
      A(LOC) = A(LOC) * (1.0-PS)                                      SAMS0907
      LOC = LOC + NR                                                  SAMS0908
   30 CONTINUE                                                        SAMS0909
   40 CONTINUE                                                        SAMS0910
      R = PROB (TSS + (NPOL-1)*TJ + NPOL*RMTTR)                       SAMS0911
      LOC = IJLOC(NP1,NP1)                                            SAMS0912
      LOC1=IJLOC(NP1,1)                                               SAMS0913
      A(LOC) = 1.0-R                                                  SAMS0914
      A(LOC1)=R                                                       SAMS0915
      NP2 = NP1 + 1                                                   SAMS0916
      NTF = 0                                                         SAMS0917
      RS = PROB (SMTTR)                                               SAMS0918
      LOC2 = IJLOC(NP2,NP2) - (NR+1)                                  SAMS0919
      LOC3 = IJLOC(NP2,1) - 1                                         SAMS0920
      DO 70 K=NP2,NSTS                                                SAMS0921
      LOC1 = IJLOC(NTF+1,K)                                           SAMS0922
      LOC2 = LOC2 + (NR+1)                                            SAMS0923
      LOC3 = LOC3 + 1                                                 SAMS0924
      IF (NTF) 60,60,45                                               SAMS0925
   45 CONTINUE                                                        SAMS0926
      T1 = TSS + (NTF-1)*TJ + NTF*RMTTR                               SAMS0927
      IF(T1-SMTTR) 60,60,50                                           SAMS0928
   50 RS = PROB(T1)                                                   SAMS0929
   60 A(LOC1) = PS                                                    SAMS0930
      A(LOC2) = 1.0-RS                                                SAMS0931
      A(LOC3)=RS                                                      SAMS0932
      NTF = NTF + 1                                                   SAMS0933
   70 CONTINUE                                                        SAMS0934
      GO TO 100                                                       SAMS0935
   80 CONTINUE                                                        SAMS0936
      DO 90 ICOL=1,NSTS                                               SAMS0937
      IJK = IJLOC(1,ICOL)                                             SAMS0938
      LROW = IJK + NR                                                 SAMS0939
      READ (IPER,1020) (A(I), I=IJK,LROW)                             SAMS0940
   90 CONTINUE                                                        SAMS0941
  100 CONTINUE                                                        SAMS0942
      IJK = IJLOC (1,ICPRO)                                           SAMS0943
      DO 110 I=1,NPOL                                                 SAMS0944
      A(IJK) = RP**(NRTSTA-(I-1))                                     SAMS0945
      IJK = IJK + 1                                                   SAMS0946
  110 CONTINUE                                                        SAMS0947
      IJ1 = IJLOC (1,1)                                               SAMS0948
      IJ2 = IJLOC (1,ICCUM)                                           SAMS0949
      A(IJ2) = A(IJ1)                                                 SAMS0950
      DO 115 I=2,NSTS                                                 SAMS0951
      IJ1 = IJ1 + NR                                                  SAMS0952
      IJ2 = IJ2 + 1                                                   SAMS0953
      A(IJ2) = A(IJ1) + A(IJ2-1)                                      SAMS0954
  115 CONTINUE                                                        SAMS0955
      WRITE (IPTR,500) SMNUM, NAME                                    SAMS0956
      WRITE(IPTR,510) RTTS                                            SAMS0957
```

```
      WRITE(IPTR,520)
      IJ = IJLOC (1,ICPRD)
      DO 120 I=1,NPOL
      IUP=NRTSTA-(I-1)
      WRITE(IPTR,530) I,IUP,A(IJ)
      IJ = IJ + 1
  120 CONTINUE
      IF (PRTMKV .EQ. 0) GO TO 132
      WRITE(IPTR,550)
      DO 130 IROW = 1, NSTS
      IJK = IJLOC(IROW,1)
      DO 125 ICOL = 1, NSTS
      RHLD(ICOL) = A(IJK)
      IJK = IJK + NR
  125 CONTINUE
      WRITE(IPTR,540) (RHLD(I),I=1,NSTS)
  130 CONTINUE
  132 CONTINUE
      IF (PCHMKV .EQ. 0) GO TO 139
      WRITE (CPNCH,1030)
      DO 137 ICOL=1,NSTS
      IJK = IJLOC(1,ICOL)
      LROW=IJK+NR
      WRITE (CPNCH,1020) (A(I), I=IJK,LROW)
  137 CONTINUE
  139 CONTINUE
      RETURN
      ENTRY SAMPRD (SMNUM,B,NR,NC,RV,IV,IST1,IST2,PROD,
     +                    KF,KS,ADJ,IFAIL,X)
      NRTNC=NR * NC
      RMTR = RV(SMNUM)
      I = 2*SMNUM
      IX = IV(I)
      NSHD = IV(I-1)
      NSTS = 2*NSHD + 1
      ICPRD = NSTS +1
      ICCUM = NSTS+2
      PROD = 0.0
      IF (IFAIL .NE. 1 ) GO TO 160
      IF (IST1 .GE. NSHD .OR. IST1 .EQ. 1 ) GO TO 140
      CALL RANDU (IX,IY,X)
      IX=IY
      XXX = PROB(TJ + RMTR)
      IF (X .LT. XXX ) GO TO 150
      IST2 = IST1 - 1
      GO TO 210
  140 CONTINUE
  150 IST2 = IST1
      GO TO 250
  160 CONTINUE
      X=-1.
      IF (IFAIL .EQ. 2) GO TO 161
      CALL RANDU(IX,IY,X)
      IX=IY
  161 CONTINUE
      IJ1 = IJLOC(1,ICCUM)
      DO 190 L=1,NSTS
      IF(B(IJ1) .GE. X ) GO TO 200
```

```
      IJ1 = IJ1 + 1                                                     SAMS1016
  190 CONTINUE                                                          SAMS1017
  200 CONTINUE                                                          SAMS1018
      IST2 = L                                                          SAMS1019
      IF (IST1 .EQ. IST2) GO TO 180                                     SAMS1020
  210 CONTINUE                                                          SAMS1021
      IJ1 = IJLOC (IST2,1)                                              SAMS1022
      IJ2 = IJLOC (I,ICCUM)                                             SAMS1023
      B(IJ2) = B(IJ1)                                                   SAMS1024
      DO 170 I=2,NSTS                                                   SAMS1025
      IJ1 = IJ1 + NR                                                    SAMS1026
      IJ2 = IJ2 + 1                                                     SAMS1027
      B(IJ2) = B(IJ1) + B(IJ2-1)                                        SAMS1028
  170 CONTINUE                                                          SAMS1029
  180 CONTINUE                                                          SAMS1030
      IF(IFAIL .GE. 1 .OR. IST2 .GT. NSHD) GO TO 250                    SAMS1031
      IJ1 = IJLOC (IST2,ICPRO)                                          SAMS1032
      PROD = B(IJ1) + ADJ*(KF-KS)*B(IJ1)                                SAMS1033
  250 CONTINUE                                                          SAMS1034
      I=2*SNNUM                                                         SAMS1035
      IV(I)= IX                                                         SAMS1036
      RETURN                                                            SAMS1037
  500 FORMAT ('1',///,5X,'PRODUCTION UNIT ',I5,5X,20A4)                 SAMS1038
  510 FORMAT(//,6X,'ROTARY  MTBF',3X,F10.3,5X,'ROTARY  MTTR',3X,F10.3,/,SAMS1039
     +5X,'SERIAL  MTBF',3X,F10.3,5X,'SERIAL  MTTR',3X,F10.3,//,6X,      SAMS1040
     +'TIME TO START+STOP',2X,F10.4,/,6X,'TIME TO JOG',9X,F10.4, /,6X,  SAMS1041
     +'RPM',2X,F5.1)                                                    SAMS1042
  520 FORMAT(///,17X,'OPERATING STATES',/,45X,'EXPECTED',/,6X,          SAMS1043
     +'STATE NO.',6X,'DESCRIPTION',9X,'PRODUCTION RATE',/)              SAMS1044
  530 FORMAT(5X,I5,5X,I5,' LINES OPERATING',10X,F5.0)                   SAMS1045
  550 FORMAT (///,50X,'MARKOV MATRIX',///)                              SAMS1046
  540 FORMAT (11E11.4)                                                  SAMS1047
 1000 FORMAT (4I2,2X,I5,10A4)                                           SAMS1048
 1010 FORMAT (2I2,I6,7F10.0)                                            SAMS1049
 1020 FORMAT (6F13.7)                                                   SAMS1050
 1030 FORMAT (80('*'))                                                  SAMS1051
      END                                                               SAMS1052
C         COST ANALYSIS
C         COST ANALYSIS
      SUBROUTINE COST (N,NB,TDOWN,TBUFF,INTVL,TPROD ,NDAY,TTPC,BUFF)    SAMS1053
      DIMENSION E(10),PROD(10),PRICE(10),SALV(10),YRS(10)               SAMS1054
      DIMENSION COPER(10),CMANT(10),CMATL(10),CDEPR(10),COP(10)         SAMS1055
      DIMENSION CMAIN(10),TRAW(10),TDOWN(10),TPROD(10),TOPER(10)        SAMS1056
      DIMENSION TMAIN(10),TDEPR(10),TOTAL(10),PRICB(10),SALVB(10)       SAMS1057
      DIMENSION YRSB(10),COPBO(10),CMANB(10),CDEP(10),COPM(10)          SAMS1058
      DIMENSION CMAB(10),BDEPR(10),BOPER(10),BMAIN(10),TBUFF(10)        SAMS1059
      DIMENSION BTOTA(10),EE(10,10),CINV(10),AVBUF(10),BUINV(10)        SAMS1060
      DIMENSION EF(10),BUFF(481,10)                                     SAMS1061
      COMMON IN,IO                                                      SAMS1062
      IF(NDAY-1) 999,1,999                                              SAMS1063
C          READING IN COST DATA
    1 READ(IN,800) DAYS,SHIFT,XINT,PDAY                                 SAMS1064
      READ(IN,300) (PRICE(I),I=1,N)                                     SAMS1065
      READ(IN,800) (PRICB(I),I=1,NB)                                    SAMS1066
      READ(IN,800) (SALV(I),I=1,N)                                      SAMS1067
      READ(IN,800) (SALVB(I),I=1,NB)                                    SAMS1068
      READ(IN,800) (YRS(I),I=1,N)                                       SAMS1069
      READ(IN,800) (YRSB(I),I=1,NB)                                     SAMS1070
```

```
      READ(IN,800) (COPER(I),I=1,N)                                     SAMS1071
      READ(IN,800) (COPBU(I),I=1,NB)                                    SAMS1072
      READ(IN,800) (CMANT(I),I=1,N)                                     SAMS1073
      READ(IN,800) (CMANB(I),I=1,NB)                                    SAMS1074
      READ(IN,800) (CMATL(I),I=1,N)                                     SAMS1075
      READ(IN,800) (CINV(I),I=1,NB)                                     SAMS1076
      READ(IN,800) (PROB(I),I=1,N)                                      SAMS1077
      READ(IN,800) (LF(I),I=1,N)                                        SAMS1078
      DO 100 L=1,N                                                      SAMS1079
      READ(IN,800) (EE(L,LL),LL=1,N)                                    SAMS1080
  100 CONTINUE                                                          SAMS1081
  800 FORMAT(10F8.0)                                                    SAMS1082
  999 CONTINUE                                                          SAMS1083
C         PRODUCTION UNIT COSTS
      IF(NDAY-1) 2,2,221                                                SAMS1084
    2 WRITE(IO,777)                                                     SAMS1085
  777 FORMAT('1',/////////,23X,'COST ANALYSIS',//,02X,'   UNIT COSTS ANDSAMS1086
     1 STATUS',/,2X,'   ---------------------   ',/,01X,'  UNIT   PURCHASSAMS1087
     2E  SALVAGE   EXPECTED   SHIFTS   NO. OPERATING  ',/,01X,'NUMBER   SAMS1088
     3PRICE($)  VALUE($)  LIFE(YRS)  PER DAY  DAYS PER WEEK ',/,01X,'---SAMS1089
     4---   --------  --------  ---------  -------  -------------  ')  SAMS1090
      DO 210 I=1,N                                                      SAMS1091
      WRITE(IO,676) I,PRICE(I),SALV(I),YRS(I),SHIFT,DAYS                SAMS1092
  676 FORMAT(1H ,1X,I2,6X,2F8.0,4X,F5.1,6X,F5.1,8X,F3.0)                SAMS1093
  210 CONTINUE                                                          SAMS1094
      WRITE(IO,888)                                                     SAMS1095
  888 FORMAT(//  ,01X,'  UNIT   COST OF',10X,'COST OF',12X,'COST OF RAW'SAMS1096
     1,06X,'QUALITY OF ',/,01X,'NUMBER   OPERATING($/HR)  MAINTENANCE($/SAMS1097
     2HR)  MATERIALS($/PC)  OUTPUT',/,01X,'------   ---------------  ---SAMS1098
     3-------------   ---------------   ----------  ')                  SAMS1099
      DO 220 I=1,N                                                      SAMS1100
      WRITE(IO,565) I,COPER(I),CMANT(I),CMATL(I),PROB(I)                SAMS1101
  565 FORMAT(1H ,1X,I2,8X,F6.2,12X,F6.2,11X,F6.4,12X,F5.3)              SAMS1102
  220 CONTINUE                                                          SAMS1103
  221 CONTINUE                                                          SAMS1104
C         CALCULATING COSTS PER TIME INTERVAL FOR THE PRODUCTION UNITS
      INT=INTVL-1                                                       SAMS1105
      DO 160 I=1,N                                                      SAMS1106
      CDEPR(I)=(PRICE(I)-SALV(I))/(YRS(I)*52.*DAYS*SHIFT)*PDAY          SAMS1107
      COP(I)=COPER(I)*XINT                                              SAMS1108
      CMAIN(I)=CMANT(I)*XINT                                            SAMS1109
  160 CONTINUE                                                          SAMS1110
      TOTPR=0                                                           SAMS1111
      DO 193 KR=1,N                                                     SAMS1112
      IF(LF(KR)-1.) 193,195,193                                         SAMS1113
  195 TOTPR=TOTPR+TPROD(KR)                                             SAMS1114
  193 CONTINUE                                                          SAMS1115
C         FINDING NECESSARY RAW MATERIAL INPUT
      DO 200 L=1,N                                                      SAMS1116
      XY=1.                                                             SAMS1117
      DO 300 LL=1,N                                                     SAMS1118
      IF(EE(L,LL)-1.) 300,301,300                                       SAMS1119
  301 XY=XY*PROB(LL)                                                    SAMS1120
  300 CONTINUE                                                          SAMS1121
      E(L)=1./XY                                                        SAMS1122
  200 CONTINUE                                                          SAMS1123
      TOTOE=0                                                           SAMS1124
      TOTOP=0                                                           SAMS1125
```

```
      TOTMA=0                                                           SAMS1126
      TOTRA=0                                                           SAMS1127
      TOT=0                                                             SAMS1128
C          CALCULATING TOTAL COSTS FOR THE PRODUCTION UNITS
      DO 190 I=1,N                                                      SAMS1129
      TDEPR(I)=CDEPR(I)                                                 SAMS1130
      TOPER(I)=COP(I)*INT                                               SAMS1131
      TMAIN(I)=CMAIN(I)*TDOWN(I)                                        SAMS1132
      TRAW(I)=TOTPR*E(I)*CMATL(I)                                       SAMS1133
      TOTAL(I)=TDEPR(I)+TOPER(I)+TMAIN(I)+TRAW(I)                       SAMS1134
      TOTDE=TOTDE+TDEPR(I)                                              SAMS1135
      TOTOP=TOTOP+TOPER(I)                                              SAMS1136
      TOTMA=TOTMA+TMAIN(I)                                              SAMS1137
      TOTRA=TOTRA+TRAW(I)                                               SAMS1138
      TOT=TOT+TOTAL(I)                                                  SAMS1139
  190 CONTINUE                                                          SAMS1140
      IF(NDAY-1) 3,3,708                                                SAMS1141
    3 WRITE(10,989)                                                     SAMS1142
  989 FORMAT(//  ,02X,'COST SUMMARY($)',/,02X,'---------------  ',/,01X,SAMS1143
     1' UNIT    COST OF',7X,'COST OF',6X,'COST OF',8X,'COST OF',5X,'TOTASAMS1144
     2L',/,    ' NUMBER   DEPRECIATION  MAINTENANCE  RAW MATERIALS  OPERSAMS1145
     3ATING   COST',/,     ' ------    ------------  -----------  --------SAMS1146
     4-----  ---------  --------- ')                                    SAMS1147
      DO 230 I=1,N                                                      SAMS1148
      WRITE(10,454) I,TDEPR(I),TMAIN(I),TRAW(I),TOPER(I),TOTAL(I)       SAMS1149
  454 FORMAT(1H ,1X,I2,8X,F9.2,4X,F7.2,6X,F9.2,6X,F7.2,3X,F9.2)         SAMS1150
  230 CONTINUE                                                          SAMS1151
      WRITE(10,707) TOTDE,TOTMA,TOTRA,TOTOP,TOT                         SAMS1152
  707 FORMAT(10X,'------------   ----------   ------------   ---------. -SAMS1153
     1--------',/,12X,F9.2,4X,F7.2,6X,F9.2,06X,F7.2,03X,F9.2)           SAMS1154
  708 CONTINUE                                                          SAMS1155
C          BUFFER COSTS
      DO 260 I=1,NB                                                     SAMS1156
      CDEP(I) =(PRICB(I)-SALVB(I))/(YRSB(I)*52.*DAYS*SHIFT)*PDAY        SAMS1157
      COPM(I)=COPMU(I)*XINT                                             SAMS1158
      CMAB(I)=CMANB(I)*XINT                                             SAMS1159
  260 CONTINUE                                                          SAMS1160
C          CALCULATING TOTAL COSTS FOR THE BUFFER
      DO 262 LA=1,NB                                                    SAMS1161
      XX=0                                                              SAMS1162
      DO 263 LB=1,INTVL                                                 SAMS1163
      XX=XX+BUFF(LB,LA)                                                 SAMS1164
  263 CONTINUE                                                          SAMS1165
      AVBUF(LA)=XX/INTVL                                                SAMS1166
  262 CONTINUE                                                          SAMS1167
      BUDEP=0                                                           SAMS1168
      BUOPE=0                                                           SAMS1169
      BUMAI=0                                                           SAMS1170
      BUTOT=0                                                           SAMS1171
      BINV=0                                                            SAMS1172
      DO 290 I=1,NB                                                     SAMS1173
      BDEPR(I)=CDEP(I)                                                  SAMS1174
      BOPER(I)=COPM(I)*INT                                              SAMS1175
      BMAIN(I)=CMAB(I)*TDOFF(I)                                         SAMS1176
      BUINV(I)=CINV(I)*AVBUF(I)*PDAY                                    SAMS1177
      BTOTA(I)=BDEPR(I)+BOPER(I)+BMAIN(I)+BUINV(I)                      SAMS1178
      BUOPE=BUOPE+BOPER(I)                                              SAMS1179
      BUDEP=BUDEP+BDEPR(I)                                              SAMS1180
```

```
      BUMAN=BUMAN+BMAIN(I)                                             SAMS1181
      BINV=BINV+BUINV(I)                                               SAMS1182
      BUTOT=BUTOT+BTOTA(I)                                             SAMS1183
  290 CONTINUE                                                         SAMS1184
      IF(NDAY-1) 4,4,5                                                 SAMS1185
C         PRINTING COST ANALYSIS
    4 WRITE(IO,810)                                                    SAMS1186
  810 FORMAT('1',/////////,23X,'COST ANALYSIS',//,02X,' BUFFER COSTS ANDSAMS1187
     1 STATUS',/,2X,' ------------------------ ',/,01X,'BUFFER   PURCHASSAMS1188
     2E  SALVAGE   EXPECTED   SHIFTS   NO. OPERATING  ',/,01X,'NUMBER   SAMS1189
     3PRICE($)  VALUE($)  LIFE(YRS)  PER DAY  DAYS PER WEEK ',/,01X,'---SAMS1190
     4----  --------  --------  ---------  -------  -------------  ')   SAMS1191
      DO 380 I=1,NB                                                    SAMS1192
      WRITE(IO,906) I,PRICB(I),SALVB(I),YRSB(I),SHIFT,DAYS             SAMS1193
  906 FORMAT(1H ,01X,I2,06X,F8.0,2X,F7.0,04X,F5.1,4X,F5.1,10X,F3.0)    SAMS1194
  380 CONTINUE                                                         SAMS1195
      WRITE(IO,811)                                                    SAMS1196
  811 FORMAT(//  ,01X,'BUFFER   COST OF',10X,'COST OF',12X,'COST OF',  SAMS1197
     1                 /,01X,'NUMBER   OPERATING($/HR)  MAINTENANCE($/SAMS1198
     2HR)  INVENTORY($-DAY/PC)    ',/,01X,'--------  ---------------  ---SAMS1199
     3--------------  ------------------  ')                           SAMS1200
      DO 390 I=1,NB                                                    SAMS1201
      WRITE(IO,908) I,COPBU(I),CMAIB(I),CINV(I)                        SAMS1202
  908 FORMAT(1H ,01X,I2,08X,F6.2,12X,F6.2,15X,F6.2)                    SAMS1203
  390 CONTINUE                                                         SAMS1204
      WRITE(IO,821)                                                    SAMS1205
  821 FORMAT(//  ,02X,'COST SUMMARY($)',/,02X,'---------------  ',/,01X,SAMS1206
     1'BUFFER   COST OF',7X,'COST OF',6X,'COST OF',4X,'COST OF',4X,'TOTASAMS1207
     2L',/,    ' NUMBER   DEPRECIATION  MAINTENANCE  OPERATING  INVENTORSAMS1208
     3Y  COSTS    ',/,    ' ------  ------------  -----------  --------SAMS1209
     4-  -------  ---------  ')                                        SAMS1210
      DO 395 I=1,NB                                                    SAMS1211
      WRITE(IO,905) I,BDEPR(I),BMAIN(I),BOPER(I),BUINV(I),BTOTA(I)     SAMS1212
  905 FORMAT(1H ,01X,I2,06X,F8.2,07X,F7.2,06X,F7.2,03X,F8.2,03X,F7.2)  SAMS1213
  395 CONTINUE                                                         SAMS1214
      WRITE(IO,922) BUDEP,BUMAN,BUOPE,BINV,BUTOT                       SAMS1215
  922 FORMAT(10X,'------------  -----------  ---------  ---------  -----SAMS1216
     1--',/,10X,F8.2,07X,F7.2,06X,F7.2,05X,F7.2,01X,F8.2)              SAMS1217
C         CALCULATING COST PER UNIT
    5 TTPC=(TOT+BUTOT)/TOTPR                                           SAMS1218
      IF(NDAY-1) 6,6,7                                                 SAMS1219
    6 WRITE(IO,191) TTPC                                               SAMS1220
  191 FORMAT(/         ,15X,'COST($) PER UNIT='    ,F6.3,/,15X,'-----------SAMS1221
     1------------    ')                                               SAMS1222
    7 RETURN                                                           SAMS1223
      END                                                              SAMS1224
C         PLOTTING ROUTINE
      SUBROUTINE XPLOT(INTVL,NB,BUFF,NPRNT,NDAY,UPLIM)                 SAMS1225
      DIMENSION BUFF(481,10),MPT(80),UPLIM(10),NUM(9),N1(80),N2(80)    SAMS1226
      INTEGER ZERO                                                     SAMS1227
      COMMON IN,IO                                                     SAMS1228
      DATA IPT,JPT,KPT,NPT /'*',' ','-','I'/                           SAMS1229
      DATA NUM /'1','2','3','4','5','6','7','8','9'/,ZERO /'0'/        SAMS1230
C         LOOP TO PLOT EACH BUFFER LEVEL
      DO 100 M=1,NB                                                    SAMS1231
      YSCAL=57./UPLIM(M)                                               SAMS1232
      WRITE(IO,550) NDAY                                               SAMS1233
  550 FORMAT('1',///////,43X,'DAY NO.',I2)                             SAMS1234
```

```
      WRITE(IO,500) M                                                   SAMS1235
  500 FORMAT(      /,40X,'BUFFER NUMBER ',I2,// ,40X,'BUFFER SIZE(IN THOSAMS1236
     1USANDS)',//)                                                      SAMS1237
C         SETTING UP AXIS
      DO 800 J=1,70                                                     SAMS1238
      N1(J)=JPT                                                         SAMS1239
      N2(J)=JPT                                                         SAMS1240
      MPT(J)=JPT                                                        SAMS1241
  800 CONTINUE                                                          SAMS1242
      DO 900 J=1,30                                                     SAMS1243
      KT=J*3000                                                         SAMS1244
      IT=KT/1000                                                        SAMS1245
      IF(KT-UPLIM(M)) 950,950,900                                       SAMS1246
  950 KT=KT*YSCAL                                                       SAMS1247
      MPT(KT)=NPT                                                       SAMS1248
      IT1=IT/10                                                         SAMS1249
      IT2=IT-10*IT1                                                     SAMS1250
      IF(IT1 .EQ. 0) GO TO 10                                           SAMS1251
      N1(KT)=NUM(IT1)                                                   SAMS1252
   10 N2(KT)=NUM(IT2)                                                   SAMS1253
      IF(IT2 .NE. 0) GO TO 900                                          SAMS1254
      N2(KT)= ZERO                                                      SAMS1255
  900 CONTINUE                                                          SAMS1256
      WRITE(IO,700) (N1(N),N=1,59)                                      SAMS1257
      WRITE(IO,850) (N2(N),N=1,59)                                      SAMS1258
  850 FORMAT(  13X,'0',100A1)                                           SAMS1259
      WRITE(IO,700) (MPT(J),J=1,70)                                     SAMS1260
      DO 600 KA=1,59                                                    SAMS1261
      MPT(KA)=KPT                                                       SAMS1262
  600 CONTINUE                                                          SAMS1263
      WRITE(IO,700) (MPT(N),N=1,59)                                     SAMS1264
  700 FORMAT(14X,100A1)                                                 SAMS1265
C         PLOTTING INDIVIDUAL POINTS
      IC=0                                                              SAMS1266
      DO 200 L=1,INTVL,NPRNT                                            SAMS1267
      IC=IC+1                                                           SAMS1268
      IF(IC-26) 1,1,2                                                   SAMS1269
    2 WRITE(IO,222)                                                     SAMS1270
  222 FORMAT('1',//////  )                                              SAMS1271
      IC=-8                                                             SAMS1272
    1 DO 300 MA=1,80                                                    SAMS1273
      MPT(MA)=JPT                                                       SAMS1274
  300 CONTINUE                                                          SAMS1275
      Y=YSCAL*BUFF(L,M)                                                 SAMS1276
      K=Y                                                               SAMS1277
      IF(K-0) 350,450,350                                               SAMS1278
  450 K=1                                                               SAMS1279
  350 MPT(K)=IPT                                                        SAMS1280
      KK=L-1                                                            SAMS1281
      WRITE(IO,400) KK,(MPT(N),N=1,80)                                  SAMS1282
  400 FORMAT(8X,I3,2X,'I',100A1)                                        SAMS1283
  200 CONTINUE                                                          SAMS1284
  100 CONTINUE                                                          SAMS1285
      RETURN                                                            SAMS1286
      END                                                               SAMS1287
```

APPENDIX B

DATA INPUT

Card Number	Variables	Format	Field Length	Comments
1	N, NB, NJ, INTVL, JSKIP, NINT, NPROD, NDAYS, JPLOT, NPRNT, NPLOT, IX	I5	12	Right hand justified.
Next NJ	CLCMKV, RDMKV, PCHMKV, PRTMKV, SMNUM, SMTYPE, NAME, IX KN(J,K), LIMIT(J,K) Appropriate Model Data (by Model)	4I2,2x, 2I2,1x, 13A2,4x,I5,5x, 3I2, 4x, 3F5.0		Information to handle subroutine MKVBLD in generating Markov transition matrices.
Next NB x NJ	A (K,I,J)	F8.0	N	-
Next 1	ISTAT (I,J)	I2	N	Right hand justified. Defines initial Markov state for each production unit.
Next NB x NJ	MFAIL (K,I,J)	I2	N	Right hand justified.
Next 1	BUFF (1,J)	F8.0	NB	Sets initial buffer levels.
Next 1	ULIM (I)	F8.0	NB	-

Card Number	Variables	Format	Field Length	Comments
Next 1	LOLIM (I)	F8.0	NB	-
Next 1	BPROB (I)	F8.0	NB	-
Next 1	BMTTR (I)	F8.0	NB	-
Next 1	ADJUS (I)	F8.0	N	-
Next 1	SEN (I)	F8.0	NB	-
Next NJ	SETUP (J,K)	F8.0	N	-

DATA INPUT (Continued)

Cost Analysis Input (Optional)

Card Number	Variables	Format	Field Length	Comments
Next 1	DAYS, SHIFT, XIOT, PDAY	F8.0	4	-
Next 1	PRICE (I)	F8.0	N	-
Next 1	PRICB (I)	F8.0	NB	-
Next 1	SALV (I)	F8.0	N	-
Next 1	SALVB (I)	F8.0	NB	-
Next 1	YRS (I)	F8.0	N	-
Next 1	YRSB (I)	F8.0	NB	-
Next 1	COPER (I)	F8.0	N	-
Next 1	COPBU (I)	F8.0	NB	-
Next 1	CMANT (I)	F8.0	N	-
Next 1	CMANB (I)	F8.0	NB	-
Next 1	CMATL (I)	F8.0	N	-
Next 1	CINV (I)	F8.0	NB	-
Next 1	PROB (I)	F8.0	N	-
Next 1	EF (I)	F8.0	N	-
Next N	EE (I,J)	F8.0	N	-

APPENDIX C

COMPUTER PROGRAM VARIABLES

Mainline Program

Variable	Description
ADJUS (J)	Percent of production unit J can be slowed down or sped up.
AVAB (I)	Availability of buffer I.
A (K,I,J)	For every complete unit of output, the number of items production unit J contributes (+) to or depletes (-) from buffer I during job K.
AVAM (J)	Percentage of time production unit J is operating.
AVAMM (I,J)	Availability of production unit J on day I.
AVGPR (I,J)	Average production of product unit J on day I.
AVGBU (I,J)	Average level of buffer J on day I.
AVAMB (I,J)	Availability of buffer J on day I.
BUFF (I,J)	Number of units in buffer J at the end of time interval I.
BMTTR (I)	Mean time to repair buffer I.
BPROB (I)	Probability of buffer I being operable.
BPROC (I)	Probability of buffer I being repaired in one time interval.
DOWN (I,J)	Percent of time production unit J is down due to buffer level or failure on day I.
IFAIL (I,J)	Logic parameter set to 1 if production unit J is to be shut down during time interval I due

Variable	Description
	to buffer level or failure; Set to 2 for shut down due to set-up procedures; Otherwise 0.
IBFAL (I,J)	Logic control set to 1 if buffer J has a mechnical failure during time interval I; otherwise 0.
ISTAT (I,J)	Matrix state which production unit J is in at the end of time interval I.
INTVL	Number of time intervals plus one that are to be simulated.
IX	Random seed generator for buffers.
JPLOT	Input parameter equal to 1 if a plot of all buffer levels over time is desired; otherwise 0.
JSKIP	Input parameter set to 1 if a cost analysis is desired; otherwise 0.
KFAST (I)	Logic control equal to 1 if production unit I is to be sped up during the next time interval; 0 otherwise.
KSLOW (I)	Logic control equal to 1 if production unit I is to be slowed down during the next time interval; otherwise 0.
KN (I,K)	Indicates sequence of jobs K that are to be produced on production unit I.
LIMIT (I,K)	List the production limits that production unit I is to produce for job K before changing jobs.
LOLIM (J)	Lower limit of buffer J.
MFAIL (I,J)	Logic control equal to 2 if production unit J is to shut down when buffer I fails or overfills and equal to 1 if production unit J is to shut down when buffer I fails or runs dry; otherwise 0.
N	Number of production units in process.
NB	Number of buffers in process.
NJ	Number of jobs to be produced.

Variable	Description
NDAYS	Number of days the simulation is to run.
NINT	Number of time intervals between printouts of current production units and buffer status.
NPROD	Number of time intervals between printout of a complete summary of production unit and buffer operation to the present.
NPRNT	Number of time intervals between points on the plots of buffer levels overtime.
NPLOT	Frequency in days of plots of buffer levels overtime.
PROD (I,J)	Output of production unit J during time interval I.
SEN (J)	Percentage of total buffer size to buffer J added to and subtracted from its lower and upper limits respectively. If the level of buffer J falls outside these new limits, the production units connected to buffer J are sped up or slowed down appropriately.
STDEM (I,J)	Standard deviation of the output of production unit J on day I.
STDEB (I,J)	Standard deviation of the level of buffer J on day I.
TPROD (J)	Total output of production unit J.
TFAIL (J)	Percent of time production unit J is down due to buffer levels or mechanical failures.
TBUFF (J)	Number of time intervals buffer J is down due to mechanical failures.
TDOWN (J)	Number of time intervals production unit J has zero production.
TOTPC (I)	Average cost per unit of output on day I.
TAVG (I)	Average output of production unit I over NDAYS.
TAVAM (I)	Average availability of production unit I over NDAYS.

Variable	Description
TSTDM (I)	Standard deviation of production unit I over NDAYS.
TTDAN (I)	Average down time of production unit I due to buffer failures or levels over NDAYS.
TAVGB (J)	Average level of buffer J over NDAYS.
TSTDB (J)	Standard deviation of the level of buffer J over NDAYS.
TAVAB (J)	Average availability of buffer J over NDAYS.
ULIM (I)	Upper limit of buffer I.
XX (J)	Random number generated in determining output of production unit J.
YY (J)	Random number generated in determining operationing status of buffer J.

APPENDIX D

COMPUTER PRINTOUTS

For Runs 1 through 6

PRODUCTION UNIT 1 CASE SUBMODULE

ROTARY MTBF	136.800	ROTARY MTTR	1.250
SERIAL MTBF	980.000	SERIAL MTTR	20.500

TIME TO START+STOP 2.4160
TIME TO JOG 0.6262
RPM 44.0

OPERATING STATES

STATE NO.	DESCRIPTION	EXPECTED PRODUCTION RATE
1	24 LINES OPERATING	1056.
2	23 LINES OPERATING	1012.
3	22 LINES OPERATING	968.
4	21 LINES OPERATING	924.

MARKOV MATRIX

0.8403E 00	0.1455E 00	0.1213E-01	0.6430E-03	0.2614E-04	0.1020E-02	0.0	0.0	0.0
0.0	0.8464E 00	0.1408E 00	0.1120E-01	0.5887E-03	0.0	0.1020E-02	0.0	0.0
0.0	0.0	0.8525E 00	0.1356E 00	0.1081E-01	0.0	0.0	0.1020E-02	0.0
0.0	0.0	0.0	0.8587E 00	0.1403E 00	0.0	0.0	0.0	0.1020E-02
0.1020E 00	0.0	0.0	0.0	0.8980E 00	0.0	0.0	0.0	0.0
0.4761E-01	0.0	0.0	0.0	0.0	0.9524E 00	0.0	0.0	0.0
0.4761E-01	0.0	0.0	0.0	0.0	0.0	0.9524E 00	0.0	0.0
0.4761E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.9524E 00	0.0
0.4761E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9524E 00

For Runs 1 through 6

PRODUCTION UNIT 2 PRIMER INSERT SUBMODULE

ROTARY MTBF	625.000	ROTARY MTTR	4.000
SERIAL MTBF	45.250	SERIAL MTTR	18.830

TIME TO START+STOP 2.2000
TIME TO JOG 0.0333
RPM 46.0

OPERATING STATES

STATE NO.	DESCRIPTION	EXPECTED PRODUCTION RATE
1	24 LINES OPERATING	1104.
2	23 LINES OPERATING	1058.
3	22 LINES OPERATING	1012.
4	21 LINES OPERATING	966.
5	20 LINES OPERATING	920.

MARKOV MATRIX

0.9413E 00	0.3617E-01	0.6661E-03	0.7822E-05	0.6576E-07	0.1049E-05	0.2186E-01	0.0	0.0	0.0	0.0
0.0	0.9428E 00	0.3472E-01	0.6116E-03	0.6855E-05	0.9911E-06	0.0	0.2186E-01	0.0	0.0	0.0
0.0	0.0	0.9443E 00	0.3327E-01	0.5593E-03	0.6938E-05	0.0	0.0	0.2186E-01	0.0	0.0
0.0	0.0	0.0	0.9458E 00	0.3180E-01	0.5153E-03	0.0	0.0	0.0	0.2186E-01	0.0
0.0	0.0	0.0	0.0	0.9473E 00	0.3081E-01	0.0	0.0	0.0	0.0	0.2186E-01
0.4379E-01	0.0	0.0	0.0	0.0	0.9562E 00	0.0	0.0	0.0	0.0	0.0
0.5172E-01	0.0	0.0	0.0	0.0	0.0	0.9483E 00	0.0	0.0	0.0	0.0
0.5172E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.9483E 00	0.0	0.0	0.0
0.5172E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9483E 00	0.0	0.0
0.5172E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9483E 00	0.0
0.5172E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9483E 00

```
PRODUCTION UNIT      3          BULLET SUBMODULE

 ROTARY  MTBF       75.760      ROTARY  MTTR          1.500
 SERIAL  MTBF     1111.110      SERIAL  MTTR          5.000

 TIME TO START+STOP          1.5000
 TIME TO JOG                 1.6666
 RPM   41.0


            OPERATING STATES
                                          EXPECTED
 STATE NO.       DESCRIPTION          PRODUCTION RATE

    1         24 LINES OPERATING            984.


                                                 MARKOV MATRIX


0.7278E 00 0.2713E 00 0.8995E-03
0.2835E 00 0.7165E 00 0.0
0.1813E 00 0.0        0.8187E 00
```

For Runs 1 through 6

For Runs 1 through 6

PRODUCTION UNIT 4 LOAD AND ASSEMBLE SUBMODULE

ROTARY MTBF	10000.000	ROTARY MTTR	9.000
SERIAL MTBF	28.570	SERIAL MTTR	35.120

TIME TO START+STOP 2.5000
TIME TO JOG 0.6262
RPM 35.0

OPERATING STATES

STATE NO.	DESCRIPTION	EXPECTED PRODUCTION RATE
1	24 LINES OPERATING	840.
2	23 LINES OPERATING	805.
3	22 LINES OPERATING	770.
4	21 LINES OPERATING	735.

MARKOV MATRIX

0.9633E 00	0.2315E-02	0.2666E-05	0.1958E-08	0.5755E-06	0.3440E-01	0.0	0.0	0.0
0.0	0.9634E 00	0.2219E-02	0.2444E-05	0.5755E-06	0.0	0.3440E-01	0.0	0.0
0.0	0.0	0.9635E 00	0.2123E-02	0.2765E-05	0.0	0.0	0.3440E-01	0.0
0.0	0.0	0.0	0.9636E 00	0.2029E-02	0.0	0.0	0.0	0.3440E-01
0.2446E-01	0.0	0.0	0.0	0.9755E 00	0.0	0.0	0.0	0.0
0.2807E-01	0.0	0.0	0.0	0.0	0.9719E 00	0.0	0.0	0.0
0.2807E-01	0.0	0.0	0.0	0.0	0.0	0.9719E 00	0.0	0.0
0.2807E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.9719E 00	0.0
0.2807E-01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9719E 00

DEPENDENCE MATRIX

JOB 1

BUFFER	PRODUCTION UNIT 1	2	3	4
1	2	1	0	0
2	0	2	0	1
3	0	0	2	1
4	0	0	0	0
5	0	0	0	0

CODE

0 PRODUCTION UNIT AND BUFFER NOT DIRECTLY CONNECTED
1 BUFFER INPUTS TO PRODUCTION UNIT
2 PRODUCTION UNIT INPUTS TO BUFFER

DEPENDENCE MATRIX

JOB 2

BUFFER	PRODUCTION UNIT 1	2	3	4
1	2	1	0	0
2	0	2	0	1
3	0	0	0	0
4	0	0	2	1
5	0	0	0	0

CODE

0 PRODUCTION UNIT AND BUFFER NOT DIRECTLY CONNECTED
1 BUFFER INPUTS TO PRODUCTION UNIT
2 PRODUCTION UNIT INPUTS TO BUFFER

For Runs 1 through 6

DEPENDENCE MATRIX

JOB 3

	PRODUCTION UNIT 1	2	3	4
BUFFER				
1	2	1	0	0
2	0	2	0	1
3	0	0	0	0
4	0	0	0	0
5	0	0	2	1

CODE

0 PRODUCTION UNIT AND BUFFER NOT DIRECTLY CONNECTED
1 BUFFER INPUTS TO PRODUCTION UNIT
2 PRODUCTION UNIT INPUTS TO BUFFER

BUFFER AND UNIT DISTRIBUTION TABLE

JOB 1

	PRODUCTION UNIT 1	2	3	4
BUFFER				
1	1.000	-1.000	0.0	0.0
2	0.0	1.000	0.0	-1.000
3	0.0	0.0	1.000	-1.000
4	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0

CODE

ZERO UNIT AND BUFFER NOT DIRECTLY CONNECTED
POSITIVE NO. PROPORTION OF UNIT PRODUCTION ADDED TO BUFFER
NEGATIVE NO. PROPORTION OF UNIT PRODUCTION TAKEN FROM BUFFER

For Runs 1 through 6

BUFFER AND UNIT DISTRIBUTION TABLE

JOB 2

	PRODUCTION UNIT 1	2	3	4
BUFFER				
1	1.000	-1.000	0.0	0.0
2	0.0	1.000	0.0	-1.000
3	0.0	0.0	0.0	0.0
4	0.0	0.0	1.000	-1.000
5	0.0	0.0	0.0	0.0

	CODE
ZERO	UNIT AND BUFFER NOT DIRECTLY CONNECTED
POSITIVE NO.	PROPORTION OF UNIT PRODUCTION ADDED TO BUFFER
NEGATIVE NO.	PROPORTION OF UNIT PRODUCTION TAKEN FROM BUFFER

BUFFER AND UNIT DISTRIBUTION TABLE

JOB 3

	PRODUCTION UNIT 1	2	3	4
BUFFER				
1	1.000	-1.000	0.0	0.0
2	0.0	1.000	0.0	-1.000
3	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0
5	0.0	0.0	1.000	-1.000

	CODE
ZERO	UNIT AND BUFFER NOT DIRECTLY CONNECTED
POSITIVE NO.	PROPORTION OF UNIT PRODUCTION ADDED TO BUFFER
NEGATIVE NO.	PROPORTION OF UNIT PRODUCTION TAKEN FROM BUFFER

For Runs 1 through 6

COST ANALYSIS

UNIT COSTS AND STATUS

UNIT NUMBER	PURCHASE PRICE($)	SALVAGE VALUE($)	EXPECTED LIFE(YRS)	SHIFTS PER DAY	NO. OPERATING DAYS PER WEEK
1	3999000.	0.	10.0	1.0	5.
2	690000.	0.	10.0	1.0	5.
3	2500000.	0.	10.0	1.0	5.
4	910000.	0.	10.0	1.0	5.

UNIT NUMBER	COST OF OPERATING($/HR)	COST OF MAINTENANCE($/HR)	COST OF RAW MATERIALS($/PC)	QUALITY OF OUTPUT
1	45.00	0.0	0.0097	0.992
2	33.00	0.0	0.0	0.996
3	33.00	0.0	0.0023	0.996
4	33.00	0.0	0.0027	0.999

COST SUMMARY($)

UNIT NUMBER	COST OF DEPRECIATION	COST OF MAINTENANCE	COST OF RAW MATERIALS	COST OF OPERATING	TOTAL COST
1	1538.08	0.0	1940.60	360.72	3839.40
2	265.38	0.0	0.0	264.53	529.91
3	961.54	0.0	454.64	264.53	1680.70
4	350.00	0.0	531.57	264.53	1146.09
	3115.00	0.0	2926.80	1154.30	7196.10

For Runs 1 through 6

COST ANALYSIS

BUFFER COSTS AND STATUS

BUFFER NUMBER	PURCHASE PRICE($)	SALVAGE VALUE($)	EXPECTED LIFE(YRS)	SHIFTS PER DAY	NO. OPERATING DAYS PER WEEK
1	400000.	0.	10.0	1.0	5.
2	400000.	0.	10.0	1.0	5.
3	400000.	0.	10.0	1.0	5.
4	400000.	0.	10.0	1.0	5.
5	400000.	0.	10.0	1.0	5.

BUFFER NUMBER	COST OF OPERATING($/HR)	COST OF MAINTENANCE($/HR)	COST OF INVENTORY($-DAY/PC)
1	0.0	0.0	0.0
2	0.0	0.0	0.0
3	0.0	0.0	0.0
4	0.0	0.0	0.0
5	0.0	0.0	0.0

COST SUMMARY($)

BUFFER NUMBER	COST OF DEPRECIATION	COST OF MAINTENANCE	COST OF OPERATING	COST OF INVENTORY	TOTAL COSTS
1	153.85	0.0	0.0	0.0	153.85
2	153.85	0.0	0.0	0.0	153.85
3	153.85	0.0	0.0	0.0	153.85
4	153.85	0.0	0.0	0.0	153.85
5	153.85	0.0	0.0	0.0	153.85
	769.23	0.0	0.0	0.0	769.23

COST($) PER UNIT= 0.040

For Runs 1 through 4

COST ANALYSIS

BUFFER COSTS AND STATUS

BUFFER NUMBER	PURCHASE PRICE($)	SALVAGE VALUE($)	EXPECTED LIFE(YRS)	SHIFTS PER DAY	NO. OPERATING DAYS PER WEEK
1	400000.	0.	10.0	1.0	5.
2	400000.	0.	10.0	1.0	5.
3	400000.	0.	10.0	1.0	5.
4	400000.	0.	10.0	1.0	5.
5	400000.	0.	10.0	1.0	5.

BUFFER NUMBER	COST OF OPERATING($/HR)	COST OF MAINTENANCE($/HR)	COST OF INVENTORY($-DAY/PC)
1	0.0	0.0	0.0
2	0.0	0.0	0.0
3	0.0	0.0	0.0
4	0.0	0.0	0.0
5	0.0	0.0	0.0

COST SUMMARY($)

BUFFER NUMBER	COST OF DEPRECIATION	COST OF MAINTENANCE	COST OF OPERATING	COST OF INVENTORY	TOTAL COSTS
1	153.85	0.0	0.0	0.0	153.85
2	153.85	0.0	0.0	0.0	153.85
3	153.85	0.0	0.0	0.0	153.85
4	153.85	0.0	0.0	0.0	153.85
5	153.85	0.0	0.0	0.0	153.85
	769.23	0.0	0.0	0.0	769.23

COST($) PER UNIT= 0.054

For Runs 5 and 6

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.04	470.	440.	400.	410.
2	0.06	282.	265.	277.	237.
3	0.52	86.	54.	33.	21.
4	0.09	198.	165.	135.	133.
5	0.09	200.	168.	135.	135.
	0.16	247.	218.	196.	187.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	506.	533.	483.	416.
2	456.	469.	442.	377.
3	283.	225.	177.	130.
4	399.	390.	339.	305.
5	402.	396.	339.	306.
	930.	929.	831.	721.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	46.	41.	41.	50.
2	28.	24.	29.	28.
3	9.	5.	3.	3.
4	20.	15.	14.	16.
5	20.	15.	14.	16.
	24.	20.	20.	23.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	47.	42.	19.	2.
2	66.	51.	41.	0.
3	90.	95.	91.	95.
4	73.	78.	71.	76.
5	79.	67.	77.	71.
	71.	67.	60.	49.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	23295.	21653.	11145.	13786.	9939.
2	24423.	22736.	16096.	15197.	12985.
3	24154.	25453.	10000.	363.	25089.
4	23800.	22731.	21540.	5431.	4140.
5	22933.	20678.	5766.	21160.	4303.
	23721.	22650.	12909.	11187.	11291.

For Run 1

BUFFER DEVIATION

DAY	1	2	3	4	5
1	3894.	5861.	3741.	7228.	3668.
2	2588.	4587.	6411.	6611.	237.
3	4649.	1775.	0.	1757.	2437.
4	3823.	6393.	6692.	320.	7661.
5	5988.	8806.	9551.	6984.	436.
	9631.	13298.	13324.	12162.	8850.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 1

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.04	479.	446.	416.	420.
2	0.05	326.	294.	283.	264.
3	0.04	470.	439.	411.	406.
4	0.05	388.	356.	340.	323.
5	0.05	334.	328.	312.	297.
	0.05	399.	373.	352.	342.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	506.	533.	486.	417.
2	466.	487.	445.	390.
3	498.	535.	485.	416.
4	495.	507.	467.	399.
5	469.	501.	458.	401.
	1089.	1147.	1047.	905.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	47.	41.	42.	51.
2	33.	27.	29.	31.
3	47.	40.	42.	50.
4	38.	33.	35.	41.
5	34.	30.	32.	35.
	40.	34.	36.	42.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	45.	38.	19.	2.
2	51.	55.	46.	0.
3	45.	34.	18.	30.
4	57.	58.	29.	20.
5	55.	66.	49.	5.
	51.	50.	32.	11.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	22882.	21083.	15614.	9195.	10000.
2	21950.	23683.	10000.	22121.	10000.
3	21533.	18576.	12249.	3138.	11227.
4	22805.	23612.	2540.	14116.	10000.
5	22170.	24037.	8634.	22373.	10000.
	22268.	22208.	9808.	14189.	10245.

For Run 2

BUFFER DEVIATION

DAY	1	2	3	4	5
1	4239.	6242.	7807.	2251.	0.
2	5898.	3290.	0.	4850.	0.
3	5471.	9182.	4182.	751.	7051.
4	5085.	4147.	2731.	6792.	0.
5	5073.	3288.	155.	4753.	0.
	11588.	12732.	9269.	9893.	7051.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 2

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.04	516.	463.	430.	423.
2	0.05	377.	326.	295.	282.
3	0.04	558.	506.	444.	454.
4	0.04	455.	411.	375.	358.
5	0.05	371.	319.	293.	267.
	0.05	455.	405.	367.	357.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	502.	535.	488.	419.
2	484.	503.	450.	396.
3	493.	535.	489.	417.
4	492.	521.	477.	409.
5	491.	498.	449.	391.
	1101.	1159.	1054.	909.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	51.	43.	44.	51.
2	38.	30.	30.	34.
3	56.	47.	45.	54.
4	46.	39.	38.	45.
5	36.	29.	30.	32.
	46.	38.	37.	43.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	42.	33.	15.	0.
2	48.	52.	39.	0.
3	35.	23.	16.	21.
4	34.	51.	22.	15.
5	52.	64.	51.	5.
	42.	45.	29.	8.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	45848.	42308.	33670.	24558.	25000.
2	46792.	46125.	25000.	44826.	25254.
3	45112.	31286.	26785.	23355.	27706.
4	30215.	46654.	11266.	31519.	25000.
5	44726.	46387.	25000.	41360.	27442.
	42539.	42552.	24344.	33124.	26080.

For Run 3

BUFFER DEVIATION

DAY	1	2	3	4	5
1	6761.	6024.	11395.	1433.	0.
2	5798.	7490.	0.	7553.	1227.
3	6711.	17180.	3031.	84.	10076.
4	18943.	6402.	9070.	10744.	0.
5	8874.	7316.	0.	12969.	6436.
	23705.	22586.	14876.	18513.	12019.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 3

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.04	496.	463.	410.	422.
2	0.06	337.	283.	285.	230.
3	0.03	607.	570.	476.	551.
4	0.05	414.	367.	357.	313.
5	0.05	355.	317.	281.	266.
	0.05	442.	400.	362.	356.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	501.	536.	485.	419.
2	468.	477.	446.	374.
3	484.	534.	491.	395.
4	493.	499.	473.	401.
5	480.	497.	444.	390.
	1086.	1139.	1047.	885.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	50.	43.	42.	51.
2	34.	26.	29.	28.
3	61.	53.	49.	66.
4	41.	35.	37.	39.
5	35.	29.	29.	32.
	44.	37.	37.	43.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	41.	35.	15.	0.
2	54.	56.	40.	0.
3	26.	16.	3.	4.
4	49.	50.	23.	19.
5	52.	64.	49.	5.
	44.	44.	26.	6.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	45843.	43355.	26070.	30592.	26551.
2	46745.	45848.	37502.	33899.	25276.
3	40262.	31590.	28808.	13783.	18301.
4	45016.	45620.	17411.	26131.	39180.
5	44136.	47317.	34319.	25531.	31296.
	44490.	42846.	28822.	25987.	28121.

For Run 4

BUFFER DEVIATION

DAY	1	2	3	4	5
1	6759.	6775.	4030.	10743.	5105.
2	6601.	7872.	8858.	10795.	1347.
3	10332.	15771.	7341.	3106.	9526.
4	8846.	6189.	10253.	15500.	4902.
5	9749.	5846.	11177.	1869.	13872.
	19217.	20780.	19459.	22030.	18305.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 4

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.05	335.	303.	271.	271.
2	0.05	329.	300.	271.	271.
3	0.05	337.	304.	271.	271.
4	0.05	362.	335.	283.	304.
5	0.05	360.	330.	300.	299.
	0.05	345.	314.	279.	283.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	477.	488.	439.	391.
2	474.	487.	439.	390.
3	476.	485.	439.	392.
4	480.	505.	445.	398.
5	483.	482.	453.	398.
	1067.	1095.	991.	881.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	33.	28.	28.	33.
2	33.	28.	28.	33.
3	34.	28.	28.	33.
4	36.	31.	29.	38.
5	36.	32.	31.	37.
	34.	29.	29.	35.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	64.	65.	19.	1.
2	55.	42.	22.	0.
3	57.	62.	32.	9.
4	61.	48.	9.	13.
5	57.	51.	0.	21.
	59.	53.	16.	9.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	23806.	24031.	11150.	13786.	10000.
2	23027.	19522.	15094.	10000.	9260.
3	22745.	23095.	10000.	15943.	12296.
4	23030.	21272.	11274.	8334.	9563.
5	23393.	22851.	7580.	8835.	16303.
	23200.	22154.	11020.	11380.	11484.

For Run 5

BUFFER DEVIATION

DAY	1	2	3	4	5
1	3939.	3295.	3738.	7228.	0.
2	4677.	7907.	7127.	0.	2608.
3	5514.	4756.	0.	6726.	4254.
4	5248.	5537.	5466.	3463.	1800.
5	3656.	4447.	3780.	3070.	6708.
	10426.	12124.	10437.	10904.	8552.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 5

AVERAGE UNIT PRODUCTION/INTERVAL

DAY	COST/UNIT($)	1	2	3	4
1	0.05	377.	324.	271.	271.
2	0.05	376.	323.	271.	271.
3	0.05	376.	324.	271.	271.
4	0.05	445.	395.	297.	344.
5	0.05	386.	334.	297.	281.
	0.05	392.	340.	281.	288.

PRODUCTION DEVIATION

DAY	1	2	3	4
1	487.	497.	439.	392.
2	486.	499.	439.	391.
3	490.	498.	439.	392.
4	493.	517.	452.	411.
5	490.	496.	452.	392.
	1094.	1121.	993.	885.

PRODUCTION UNIT AVAILABILITY

DAY	1	2	3	4
1	38.	30.	28.	33.
2	38.	30.	28.	33.
3	37.	30.	28.	33.
4	45.	37.	31.	41.
5	38.	31.	31.	34.
	39.	31.	29.	35.

UNIT DOWNTIME DUE TO BUFFERS

DAY	1	2	3	4
1	56.	57.	15.	0.
2	49.	45.	13.	0.
3	60.	60.	32.	6.
4	47.	40.	5.	6.
5	55.	59.	0.	22.
	54.	52.	13.	7.

AVERAGE BUFFER SIZE

DAY	1	2	3	4	5
1	46740.	47029.	26224.	30592.	25000.
2	44963.	40198.	31673.	25000.	22632.
3	47154.	47707.	25000.	34780.	29296.
4	46244.	41791.	24780.	22353.	22477.
5	45309.	46980.	17295.	20966.	42288.
	46980.	44593.	24994.	26738.	28338.

For Run 6

BUFFER DEVIATION

DAY	1	2	3	4	5
1	6992.	5592.	3873.	10743.	0.
2	7813.	11557.	11083.	0.	5406.
3	7280.	4684.	0.	10994.	7146.
4	7751.	9525.	5395.	5996.	6672.
5	3998.	6655.	9993.	8494.	10223.
	17435.	17938.	16334.	18557.	15143.

BUFFER AVAILABILITY

DAY	1	2	3	4	5
1	94.	100.	100.	100.	100.
2	100.	100.	100.	100.	100.
3	100.	100.	85.	100.	100.
4	97.	100.	95.	100.	100.
5	100.	95.	100.	100.	100.
	98.	99.	96.	100.	100.

For Run 6

REFERENCES

(1) Chu, T.D., "Development of an Optimal Operating Policy for Module A1 of SCAMP Project", Unpublished Report, Product/Production Graduate Engineering Program, USAMC Intern Training Center, Red River Army Depot, 1975.

(2) Moore, F.G., Manufacturing Management, Richard D. Irwin, Inc., Homewood, Ill., 1969.

(3) Snyder, T.R., "The Simulation and Analysis of Continuous Single Product Manufacturing Systems", Unpublished Report, Product/Production Graduate Engineering Program, USAMC Intern Training Center, Red River Army Depot, 1974.

(4) Taha, H., An Introduction to Operations Research, Mac Millan Co., Inc., New York, N.Y., 1973.

(5) United States Army Material Command, "Report on the Analytical Modeling of the Bullet Submodule", Intern Training Center, Texarkana, Texas, (1973).

(6) United States Army Material Command, "Report on the Analytical Modeling of the Case Submodule", Intern Training Center, Texarkana, Texas, (1973).

(7) United States Army Material Command, "Report on the Analytical Modeling of the Primer Insert Submodule", Intern Training Center, Texarkana, Texas, (1973).