LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY

AUTHOR/FILING TITLE

FLEURY, P

ACCESSION/COPY NO

089992/01

VOL NO | CLASS MARK

ARCHIVES
COPY

FOR REFERENCE ONLY

OPERATION STRATEGIES FOR A SHOE BATCH MANUFACTURING SYSTEM

by

PAULO FERNANDO FLEURY DA SILVA E SOUZA

APPENDICES

Loughborough University
of Technology Library
Date Jan.77
Class
Acc No 089992/01

APPENDIX 1

DESCRIPTION OF THE COMPUTER PROGRAM

A1 - 1 Introduction

The computer program was written in 1900 CSL [1], an ICL package for simulation applications and ran in the Loughborough University Computer Centre.

This version of CSL allows the user to provide his own statements and routines in standard Fortran IV, if desired. Because the ICL package would not provide for the use of antithetic numbers, there was a need to introduce routines and statements in Fortran within the CSL program. Those statements are indicated by an F in the first column of a statement.

The description of the simulation program will be made in four parts:

i) An explanation about the components (segments) of the program (main program and subroutines)

ii) A list with explanations of the major variables in the program

iii) A series of flow charts corresponding to the main program and subroutines £ALLC, £BLLC and £STCT

iv) The full listing of the main program and all subroutines

(1) - CSL manual is edited by ICL, as technical publication 3386.

A1 - 2 Components (segments) of the program

The CSL program is composed of a main program (MASTER MACH) and three subroutines (£ALLC , £BLLC and £STCT). Subroutines £ALLC and £BLLC are both used to sequence the jobs in queue and to load them into the machines. The difference between them relates to the fact that £ALLC is used for the priority rules which are designed to avoid setup times (FIFOM, FIFOMB, SLACKM and SPTM), while £BLLC is used for the priority rules which do not avoid setup times (FIFO, FIFOB, SLACK and SPT). Subroutine £STCT is used to control the inventory when it is switched-on.

Apart from those four CSL routines which were written by the author, a series of Fortran subroutines and functions were incorporated into the program in order to allow the generation of antithetic random numbers. Those routines belong to the NAG library of subroutines (2), subgroup G05. They were slighly modified in order to serve the purposes of this study.

Below is a list of the Fortran routines adapted from the NAG library with their modified names and corresponding original names (NAG library names).

Modified name	Original NAG name (2)	OBJECTIVE
FADF	G05 ADF	To generate a random number from a standard normal distribution

(2) For description of those programs see the NAG LIBRARY MANUAL (Mark 4), NAG OXFORD (Subgroup G05).

(cont.)

Modified name	Original NAG name	OBJECTIVE
FAEF	G05 AEF	To generate a random number from a normal distribution with mean AM and standard deviation S
FABF	G05 ABF	To generate a random number from a uniform distribution in the range A, B
FACF	G05 ACF	To generate a random number from a exponential distribution with mean A
FAFF	G05 AFF	To generate a random number from a lognormal distribution with mean A and standard deviation B
FARF	G05 ARF	To determine the reference vector in a discrete distribution
FATF	G05 ATF	To determine the cumulative distribution for the discrete uniform distribution
FAVF	G05 AVF	To determine the cumulative distribution for the Poisson distribution
FAWF	G05 AWF	To determine the cumulative distribution of a histogram provided by the user
FAZF	G05 AZF	To return a pseudo random number from a required distribution

Apart from these 10 Fortran routines whose detailed description can be found in the NAG manual, there is an extra routine provided by the author, which allows the generation of antithetic random number. This routine is called SUBROUTINE MYRAN (P, NSTR) and works by generating a random number from the CSL internal random generator (Subroutine

IRAN1 in the Fortran translation), and calculating a random number as a function of the value of P and NSTR. When P = 1 the routine generates a stream of 'straight' random number, but if P = -1, the routine generates an antithetic stream. The value of NSTR is used to indicate which stream of random numbers should be generated.

The modifications introduced in the 10 NAG routines consisted in creating a new argument (NSTR) to take account of the different streams, and generating the random streams from MYRAN, instead than from G05 AAF which is the internal random generator for the NAG library.

The main program and all the subroutines are stored in a filing system, in binary version, with all the data also stored in another file. In its present version the program runs in the following mode.

i) The data which is on file is edited

ii) The program reads those data and executes three antithetic pairs of runs in accordance with the sampling procedure described in chapter 5

iii) The results of each antithetic pair are output through the line printer and to a file for posterior statistical analysis

iv) If more than 3 pairs of antithetics is desired, the program can be submitted again such that in each submission the output of the program is stored in a different file which can be accessed later such that all antithetic pairs from different runs are put together for statistical analysis.

A1 - 3 List of variables

In order to facilitate reference the variables will be divided into two groups. The first group is composed of the definition variables (entities) which constitutes the major structure of the model. These variables will be listed in order of their declaration in the main program (they are equivalent to their corresponding variables in the subroutines).

The second group of variables is composed of the input variables whose values are read as data. All data is read from the main program. These variables are going to be listed in accordance with the sequence in which they are input to the model (read).

A1 - 3.1 Definition variables (Entities)

1. CLASS TIME ORDER.300 (3) SET CUSTQUEUE BSET POOLB

   The entity ORDER represents the customer's orders. During the initialization all 300 entities are loaded in POOLB. It is assumed that no more than 300 ORDERS will be in the system at any one time. In case this happens the program is halted with a message:'POOLB IS NOT ENOUGH'.

   Each ORDER has three attributes:

   ORDER.I(1) = The product style required

   ORDER.I(2) = The number of different product sizes required

   ORDER.I(3) = The total quantity ordered

   (These three attributes do not need be initialized).

2. CLASS JOB. 900 (4) SET INQUEUE, ATQUEUE BSET POOL

Entities JOB represent the production batches for individual product sizes. At the initialization of each run all entities JOB are loaded in POOL. It is assumed that there will be no more than 900 entities JOB at the shop at any time. In case this number is reached the program is halted with a message: 'POOL IS NOT ENOUGH'. The sets 'INQUEUE' and 'ATQUEUE' are both used to handle JOBS arriving in the shop. The former is used during the arrival time of 'ORDERS' and the latter to keep the JOBS which are waiting to be loaded at the machines.

Each JOB has four attributes:

JOB.I(1) = Identification numberof the ORDER to which the JOB belongs

JOB.I(2) = The product size

JOB.I(3) = The 'JOB' batch size

JOB.I(4) = The code number of the product represented by this JOB. This code number is used to identify the moulds which are able to manufacture the particular product represented.

(Those values do not need to be initialized).

3. CLASS TIME MACHINE . 2(2) BSET IDLE, BUSY

Entities MACHINE represent individual multiple stations machines At the initialization of each run all machines which are in operation are loaded into set 'IDLE'. Each machine has two attributes:

MACHINE . I(1) An on-off attribute to indicate whether or not the machine is switched on in an experiment. This value needs to be input at the initialization phase (1 = ON), (0 = OFF)

MACHINE . I(2) = A parameter used to indicate the number of stations loaded in a given instant of time. This variable is set at zero during the initialization phase.

If a machine has all its stations loaded it is taken from set IDLE, otherwise it remains in IDLE and BUSY. It is taken from BUSY only when no station is loaded.

4. CLASS TIME STATION . 24(5) BSED LOADED.2, ULOADED.2

The entities 'STATION' represent the individual stations in the multiple stations machines. In the initialization of each experiment, 'STATIONS' numbers 1 to 12 are loaded into ULOADED.1, meaning that they belong to MACHINE.1, while 'STATIONS' 13 to 24 are loaded into ULOADED.2.

The attributes of each 'STATION' are:

STATION.I(1) = The identification number of the JOB which is loaded at the STATION

STATION.I(2) = Identification number of the mould which is mounted at the STATION

STATION.I(3) = A parameter used in order to cope with the changing value of 'process cycle time'. It keeps the value of the quantity still left to be produced from each JOB at a given point in time

STATION.I(4) = A parameter used to keep a time record related to the instant of times in which STATION.I(3) is measured

STATION.I(5) = A dummy parameter

Attributes STATION.I(1) and STATION . I(2) must be initialized. STATION . I(1) is set to zero and STATION . I(2) is set to the identification number of a mould. It is assumed that there is always a mould mounted in each station.

5. CLASS TIME STYLE .3(4) BSET POOLA

Entities STYLE represent individual product styles. Their time cells (T.STATION.I) are used to keep the value of arrival time of the next ORDER at the system. At the initialization period all entities STYLE are loaded at POOLA.

The attributes of STYLE are the following:

STYLE . I(1) = The number of product sizes in the product range

STYLE . I(2) = Total quantity to be demanded in the next customer order

STYLE . I(3) = Smallest product size in the product style range

STYLE . I(4) = Largest product size in the product style range

All four attributes must be initialized at the beginning. The values of STYLE . I(1) and STYLE . I(2) change during a run, while STYLE . I(3) and STYLE.I(4) are maintained constant. The time cells must also be initialized.

6. CLASS MOULD. 45(3) BSET PREE

Entities MOULD represent the moulds available in the system. If a MOULD is not mounted in a station, it is in 'PREE'. At the

beginning of each run all 'MOULDS' which are not mounted in a station must be loaded in 'PREE'. The three attributes of MOULD contain the code number of the product sizes which can be manufactured by that particular MOULD. A MOULD can be used in the manufacture of up to 3 different product styles. The values of the attributes are read as data at the beginning of each experiment.

A1 - 3.2 The input variables

1. NUEXP — Used to indicate the number of the experiment being conducted

2. PRIOR — Indicates whether to use subroutine £ALLC or £BLLC (PRIOR = 1,use £ALLC; PRIOR = 0, use £BLLC)

3. BOF — Used in conjunction with PRIOR, to indicate whether to use FIFO; FIFOB; FIFOM or FIFOMB priority rules, in accordance with the following table:

PRIOR	BOF	PRIORITY RULE
0	0	FIFO
0	1	FIFOB
1	0	FIFOM
1	1	FIFOMB

4. SOF — Used in conjunction with PRIOR to indicate whether to use SPT or SPTM rules. When SOF = 1, either SPT or SPTM is used depending on whether PRIOR is zero or one. When SOF = 1, BOF and SLACK must be set to zero.

5. SLACK — Used in conjunction with PRIOR to indicate whether to use SLACK or SLACKM priority rules. When SLACK = 1, either SLACK or SLACKM are used depending on whether PRIOR is zero or one; when SLACK = 1, BOF and SOB must be zero.

6. EMPIRIC — Controls whether to use empirical or theoretical distributions of interarrival times and demand. If EMPIRIC = 1, use empirical date; if EMPIRIC = 0, use theoretical distributions.

7. STATISTIC — Used to establish the length of the stabilization period for each run, which is measured by the number of completed orders

8. TSIMUL — A dummy variable

9. STARPOINT — A dummy variable which was used to control the output of messages from the program. Its value is set in terms of simulation time units. If the simulation time is greater than STARPOINT the program outputs messages which are used as diagnostic information. It must be unitialized with a large number (preferably above 4000 000), if the output is not desired.

10. NMOULD — Indicates the number of moulds to be used in a particular experiment

11. NRUN — Indicates the number of subdivisions if the run subdivision method of xampling is used. If independent runs is used, NRUN = 1

12. SZRUN — Indicates the length of a run in terms of number of completed orders

13. TIRUN — Indicates the number of replication runs. A pair of antithetic runs correspond to a value of 2 for TIRUN

14. NDAYS — Indicates the number of working days per week. During the study NDAYS was set to five

15. NMACHB — Indicates the number of machines defined in the program; NMACHB = 2

16. VCONV — Indicates the number of minutes per working days

17. MAXLOT — Used to set the value of the parameter for the splitting procedure. It indicates how large a job should be before it is split into smaller batches

18. NSTYL — Indicates the number of product styles (NSTYL =3)

19. GENSTR — A value used to initialize the different streams of random numbers for the different runs. If more than one submission for the same experiment is made, the difference between the values of GENSTR must be at least 100 to avoid repetition of the same stream

20. NIRUN — Is a counter to control the number of independent runs. Used in connection with TIRUN. Initialized at 1

21. MXMOULD — Indicates the maximum number of moulds defined in the program. MXMOULD = 45

22. IPARM (I) — Used to hold the values of any integer parameters of the theoretical probability distributions. It has a dimension of 20. In the present version of the program all values of IPARM = 0, as no integer parameters are used

23. PARM (I) Used to hold the values of real parameters of the theoretical probability distributios.

PARM (1) to PARM (3) hold the mean values of the interarrival times for each of the three product styles. The unit used for the interarrival times is hours. A typical value is 42.0 hours for the case of a load factor of 85%, the average size of order equals 1600, and 1 machine.

PARM (4) to PARM (6) are dummy parameters and set to zero.

PARM (7) to PARM (9) are used to keep the mean value of the distributuions of order size. Typical values are 1600.0 and 1000.0

PARM (10) to PARM (12) are dummy parameters and set at zero

PARM (13) and PARM (14) are used respectively to store the values of the mean and standard deviation of the (normal) distribution of setup times. The unit used is minutes. Typical values used were 8.0 and 16.0 for mean values and 1.5 and 3.0 for standard deviation

PARM (15) to PARM(20) are dummy parameters and set at zero.

24. CUSTARA (I,J), CUSTARB (I,J), CUSTARC (I,J) Are used to keep the values of the empirical distributions of interarrival times for each of the three product styles. All three distributions have a dimension (2, 9). The unit used is hours

25. VOLDA (I,J)
VOLDB (I,J)
VOLDC (I,J)

Are used to keep the values of empirical distributions of order size for each of the three product styles. The dimension of all three distributions is (2,9)

26. MACSET (I,J)

Is used to keep the values of the empirical distribution of setup times. The unit used is minutes. The dimension of variable is (2,7)

27. TDISA (I,J),
TDISB (I,J),
TDISC (I,J)

Are used to keep the values of the empirical distributions of probabilities which are used to select the distributions of proportions of demand for the different product sizes in a range. Each array corresponds to a product style.

Dimensions of each array is (2,4).(Used with empiric distributions)

28. RTYLA (I,J,K)

Array used to keep the various distributions of proportions for each of the three product styles. Dimension is (3,3,15)

29. SSTYLA (I,J)

A control variable used to indicate whether or not a particular product style should have its job split into smaller batches. Used in conjunction with MAXLOT. If SSTYLA (i,j) = 1, product style i, size j is considered for splitting, but if SSTYLA (i,j) = 0, then the product is never considered for splitting. Dimension of the array is (3,15)

30. STOCK (I,J)

A control variable used to indicate whether or not a particular product size is kept in stock. If STOCK (i,j) = 1, product (i,j) is kept in stock. If STOCK (i,j) = 0, product (i,j) is not kept in stock. Dimension for this variable is (3,15)

31. RPOINT (I,J) A variable used to keep the record of the reorder point for each one of the product sizes. Dimension is (3,15)

32. EBQ (I,J) Variable used to keep the values of the 'replenishment orders' batch size, for each of the product styles. Dimension is (3, 15)

33. MOULD.I(J) Attributes of 'MOULD' relating to the product sizes it can manufacture. It has been defined already. Dimension (45, 3)

34. BCMACHINE(I) Variable used to keep the value of the fixed part of the 'process cycle time' of the stations of a particular machine (BCMACHINE = N x MC in equation 3.1 of paragraph 3.3.1, chapter 3. Dimension of BCMACHINE is (2)

35. FACHINE (I) Variable used to keep the value of the variable part of the 'process cycle time' (FACHINE = 'I' in equation 3.1 of paragraph 3.3.1, chapter 3). Dimension of FACHINE (I) is (2)

36. RVOL (I,J) A correction factor used to adjust the value of the order size in function of the pattern of demand (proportions) generated. Dimension of this variable is (3,3)

37. ARATE A correction factor used to modify the interarrival time. It must be set to 1 as it is not used anymore

38. VRATE A correction factor used to modify the value of size of orders. It must be set to 1 as it is not used anymore

39. PROCUM (I) A Fortran variable used to store the probability distributions of the pattern of proportions for demand. It is used with the theoretical distributions of interarrival time and total quantity for demand. Dimensions of this variable is (14)

40. PERIOD A variable used as a counter of the number of runs executed. It is initialized with a value zero

41. T.STYLE.I A time cell used to keep the value of the next arrival of an order of style I. It is initialized with any value larger than zero. The unit used is minutes. The dimension of this variable is (3)

42. STYLA (I,J) A variable used to store the values of the proportions that are required for each product size in a given order. It can be initialized with any of the patterns of proportions. Its dimension is (3,15)

43. STYLE.I(Z) Represent the attributes of the entities CLASS TIME STYLE as defined before. It must be initialized at the beginning of each run to define the characteristics of the first order arriving at the shop for each product style. Dimension of this variable is (3,4)

44. QTSTCK (I,J) Variable used to keep the record of stock level for each individual product size. It can be initialized with any desired positive value. The dimension of this variable is (3,15)

45. STATION.I(J) Represent the attributes of the entities CLASS TIME STATION, as described before. Its dimension is (24,4) representing 4 attributes for each of the 24 sta-

tions. Only attributes 1 and 2 are initialized. STATION.I(1) is set to zero, indicating that no job is loaded at STATION.I and STATION.I(2) is set to the code number of different moulds. Each STATION.I(2) must be initialized with a unique code number

46. MACHINE.I(J) As described before this variable represent the attributes of the entities CLASS TIME MACHINE. The dimension of this variable is (2,2). MACHINE. I(1) must be initialized with 1 or zero to indicate respectively that the machine is switched-on or off in the experiment. MACHINE.I (2) is set to zero at the initialization or each run to indicate that none of the stations in the machine are loaded

A1 - 4 The flow charts

In the following pages five flow charts are presented. Figure A1 - 1 presents the flow chart for the procedure which is followed during the execution of pairs of antithetic runs in a single job submission. Figure A1 - 2 presents the flow chart of the main program MASTER MACH. Figure A1 - 3 presents the flow chart of the subroutine £ALLC. Figure A1 - 4 presents the flow chart of the subroutine £BLLC. Finally, figure A1 - 5 presents the flow chart of the subroutine £STCT.

FIGURE A1 - 1

FLOW CHART OF PROCEDURE FOLLOWED FOR THE EXECUTION OF PAIRS OF ANTITHETIC RUNS

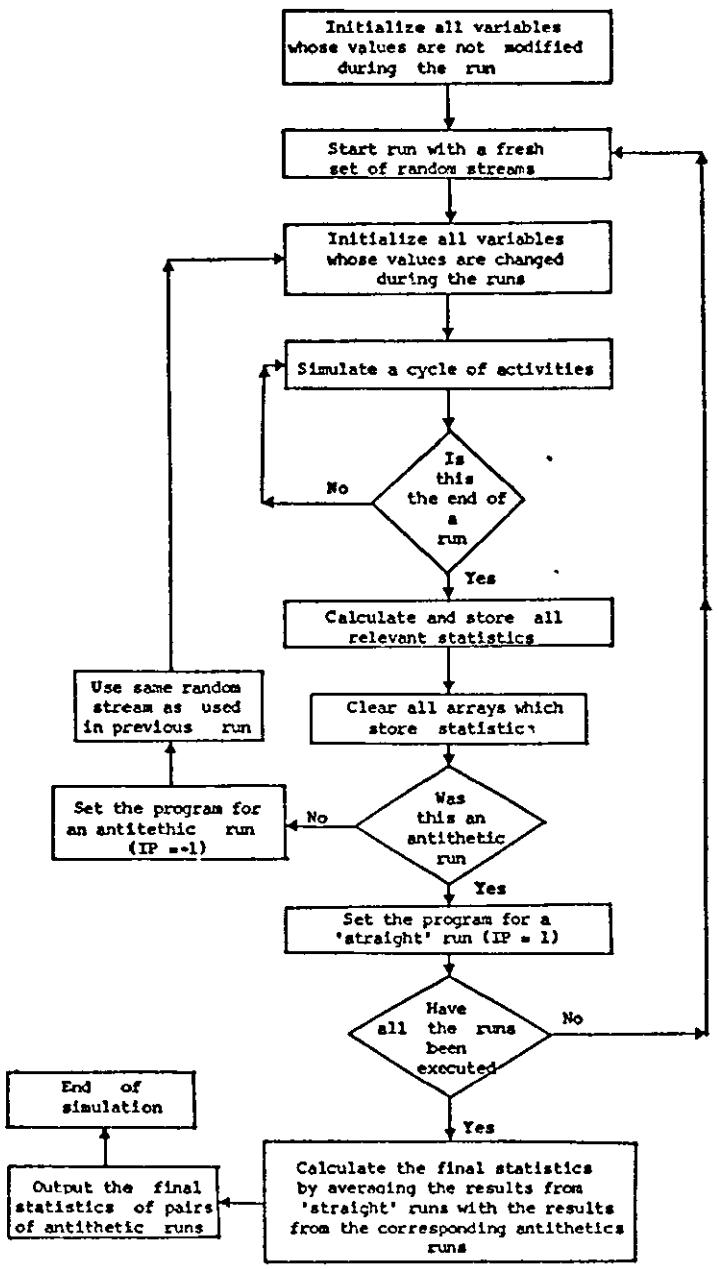


FIGURE A1 - 2

FLOW CHART OF MAIN PROGRAM

(MASTER MACH.)

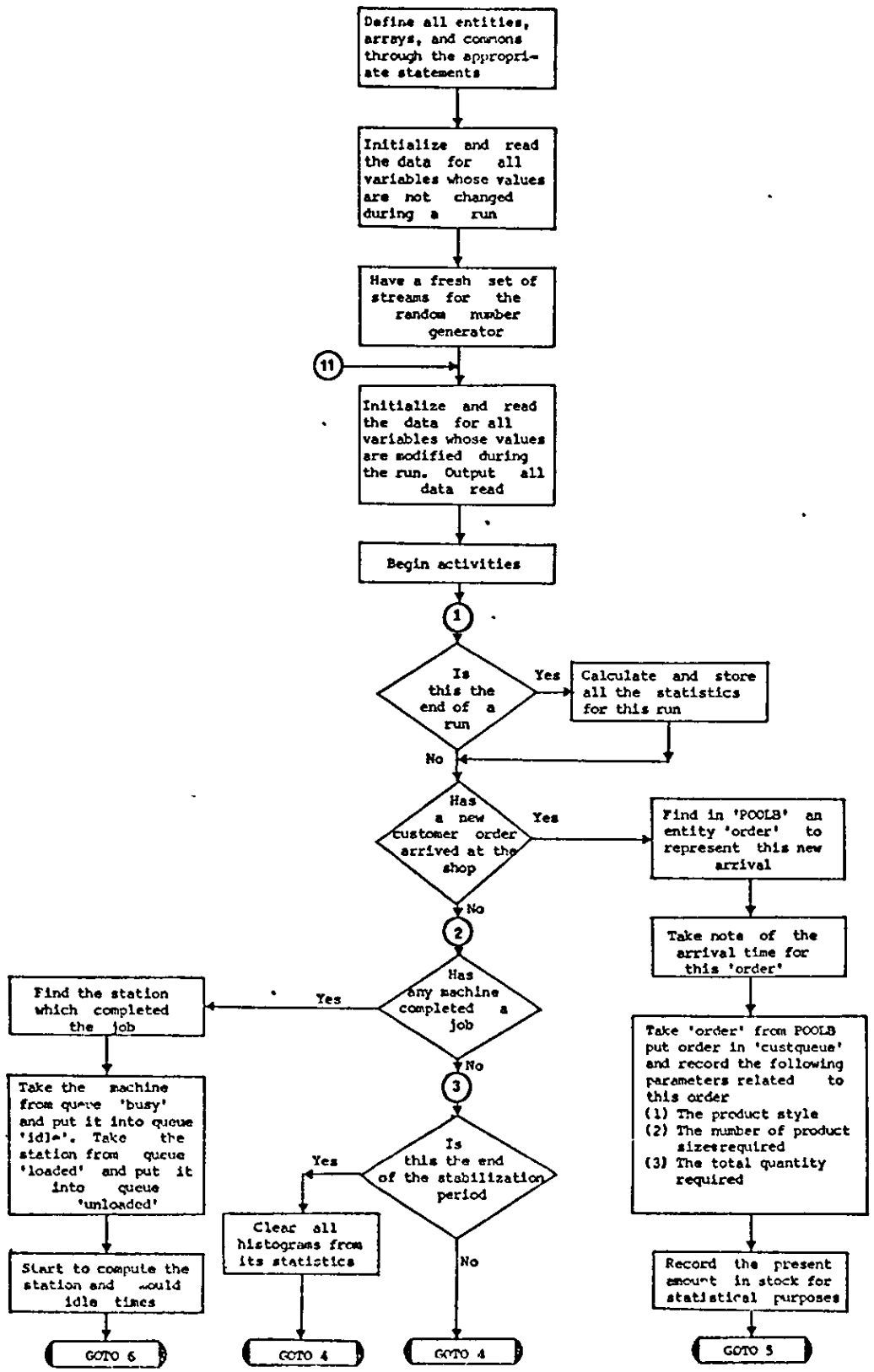


FIGURE A1 - 2 (CONTINUATION)

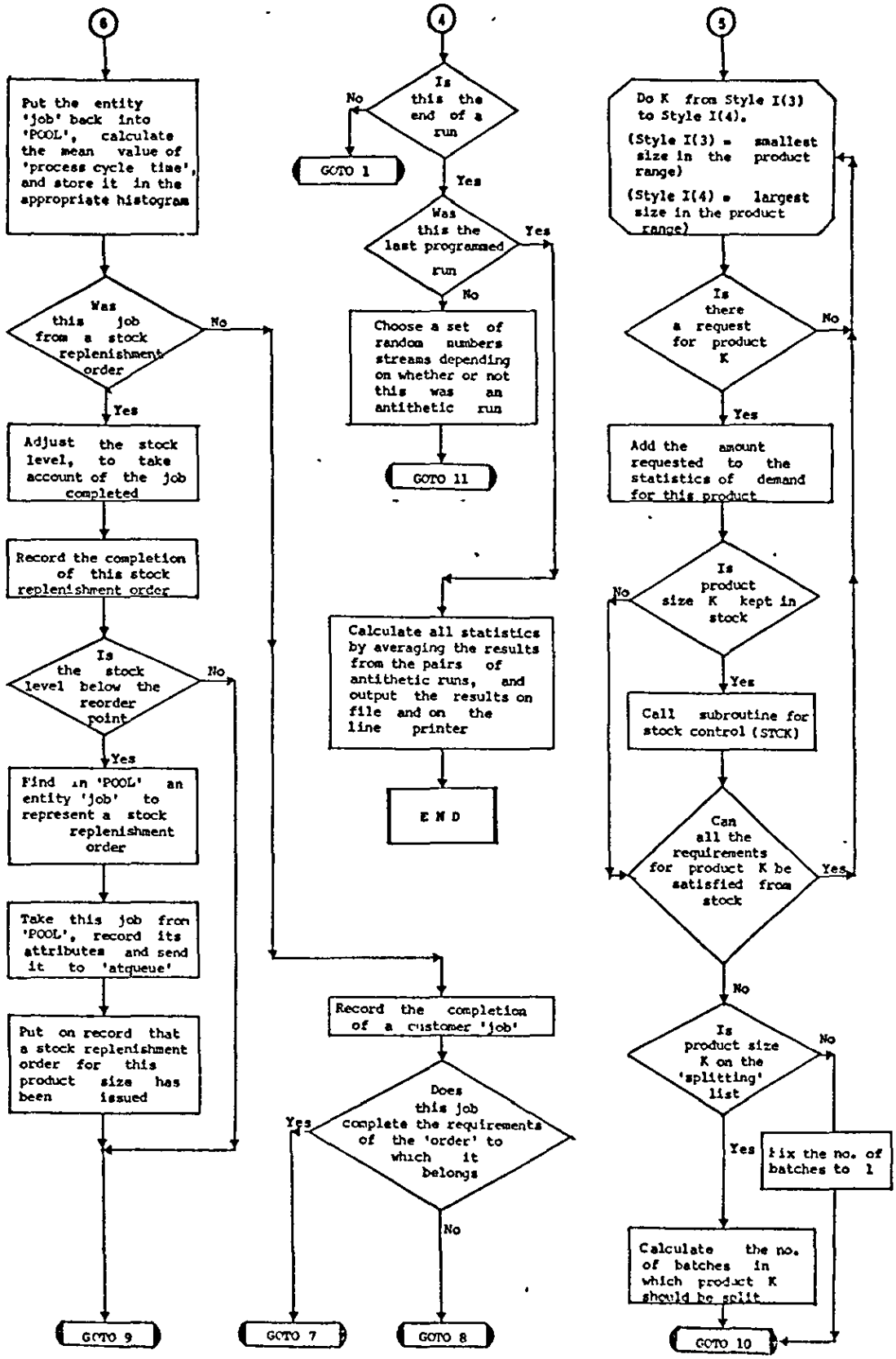


FIGURE A1 - 2 (CONTINUATION)

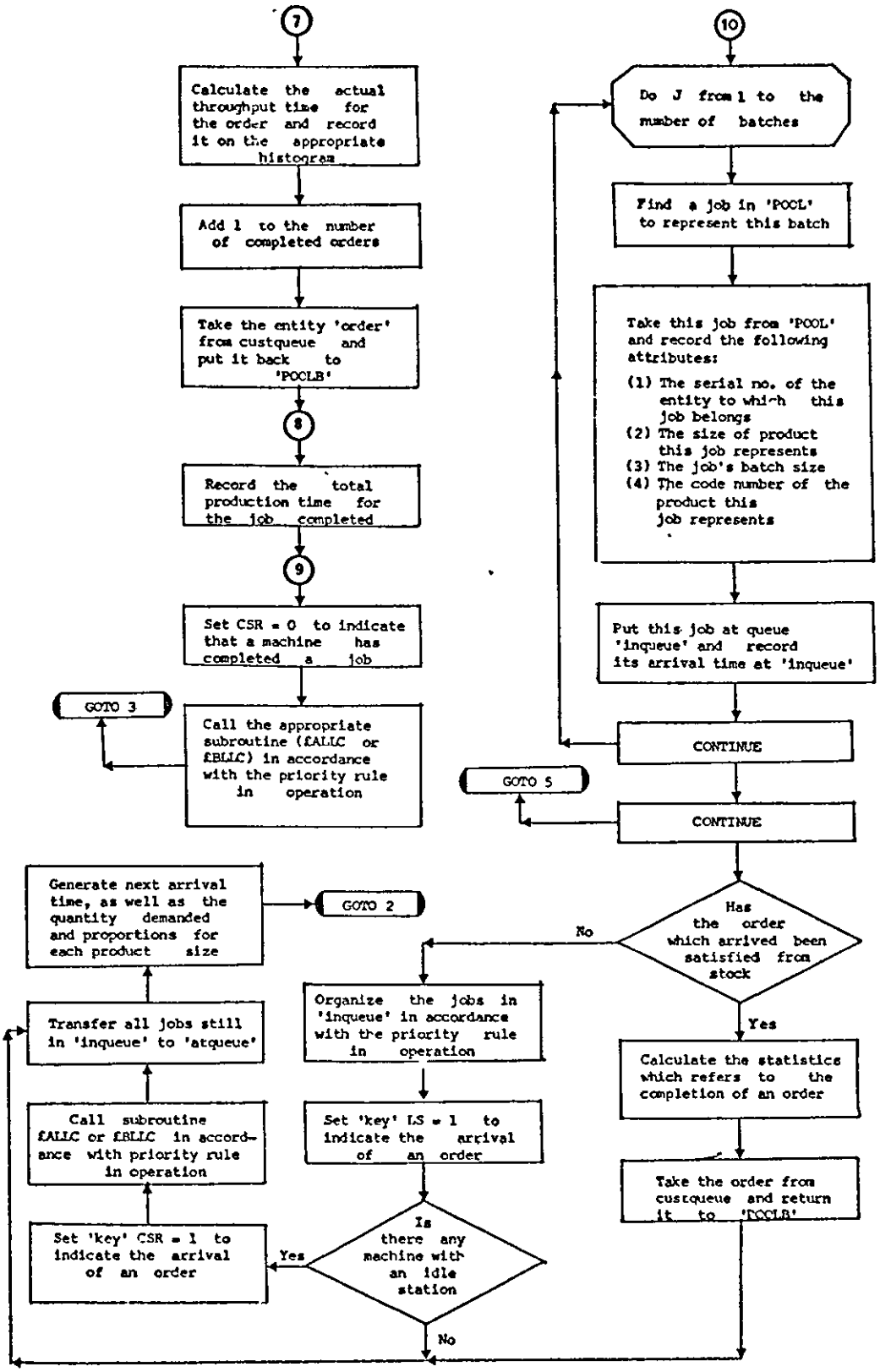


FIGURE A1 - 3

FLOW CHART OF SUBROUTINE ALIC FOR LOADING JOBS INTO MACHINES

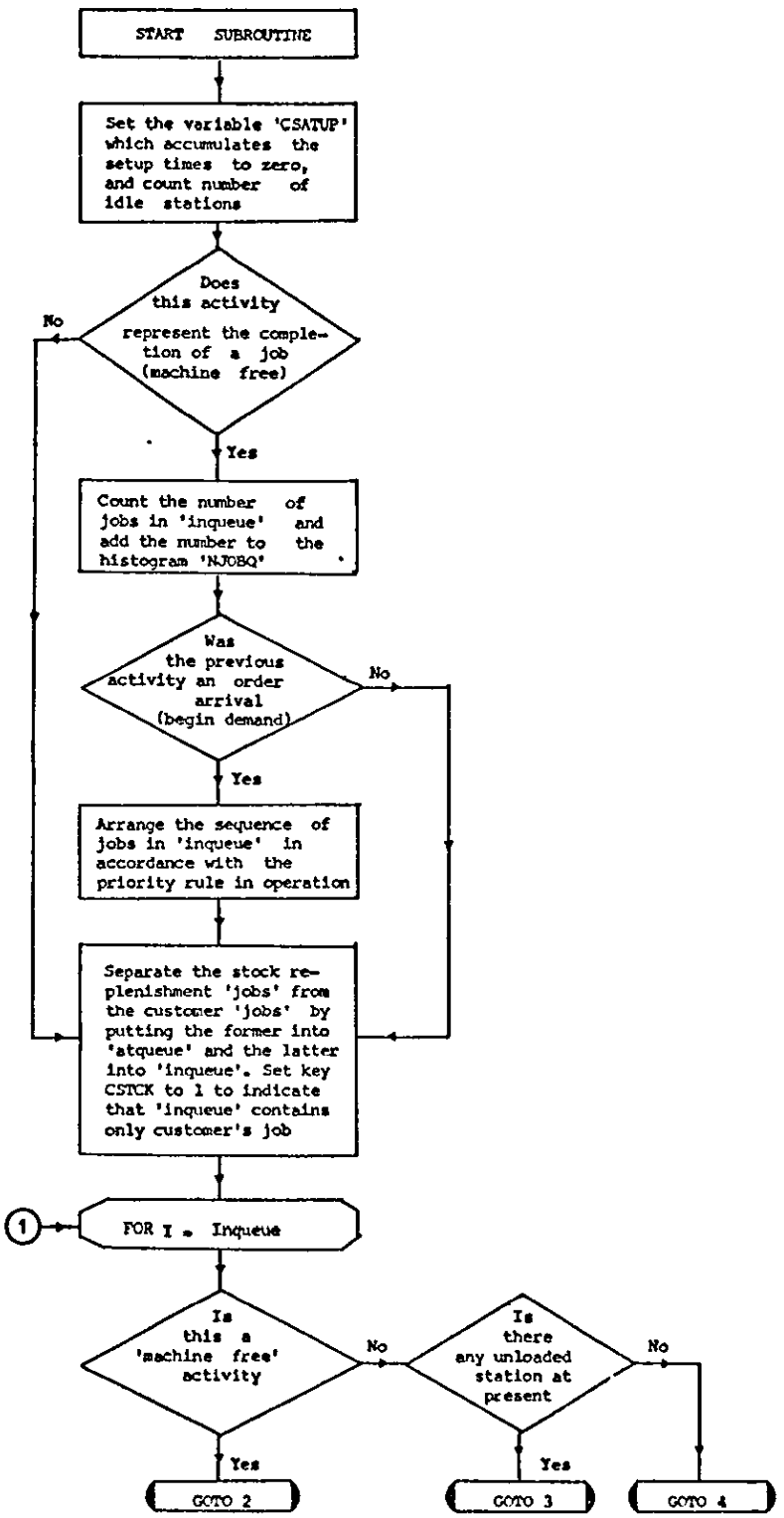


FIGURE A1 - 3 (CONTINUATION)

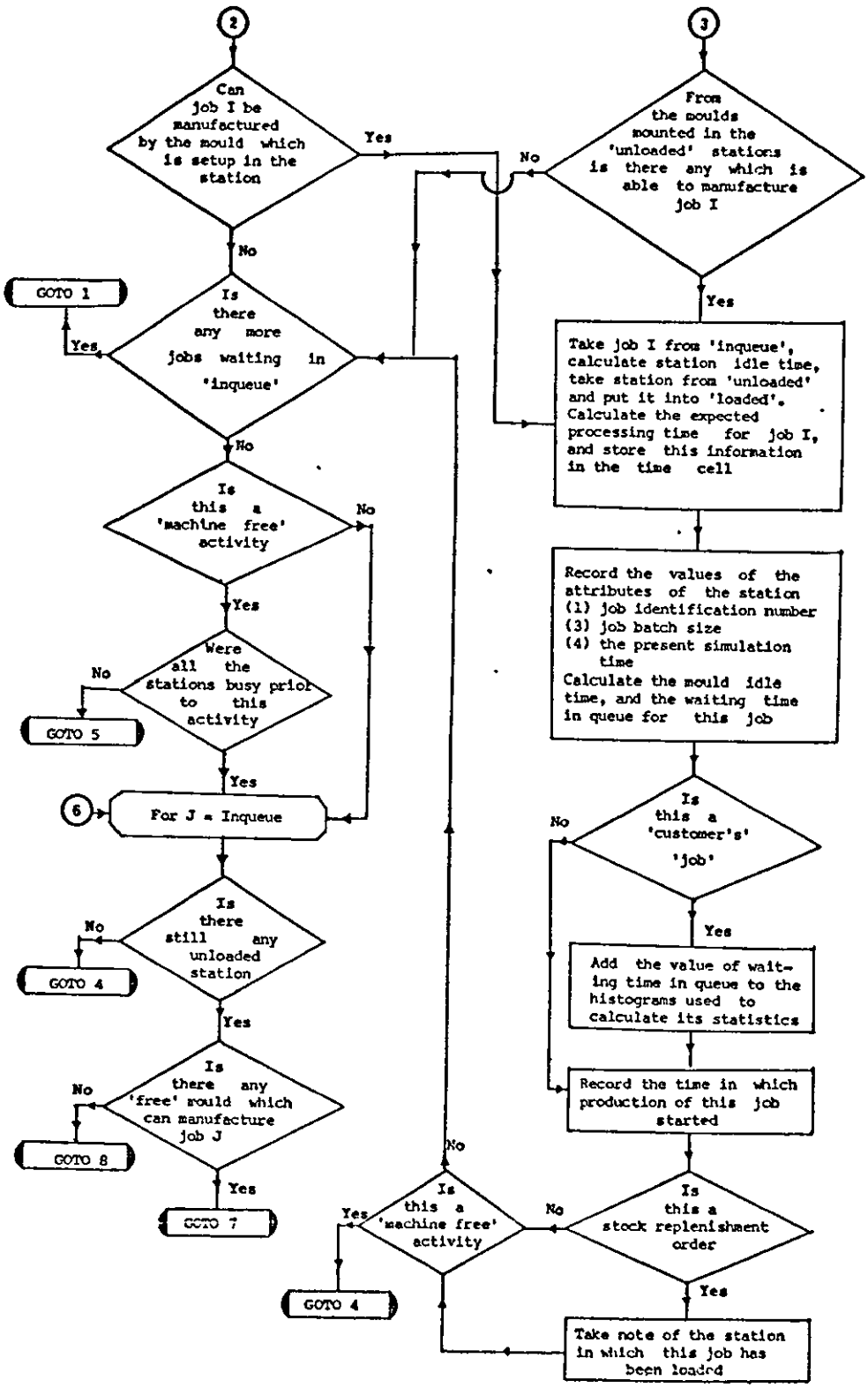


FIGURE A1 - 3 (CONTINUATION)

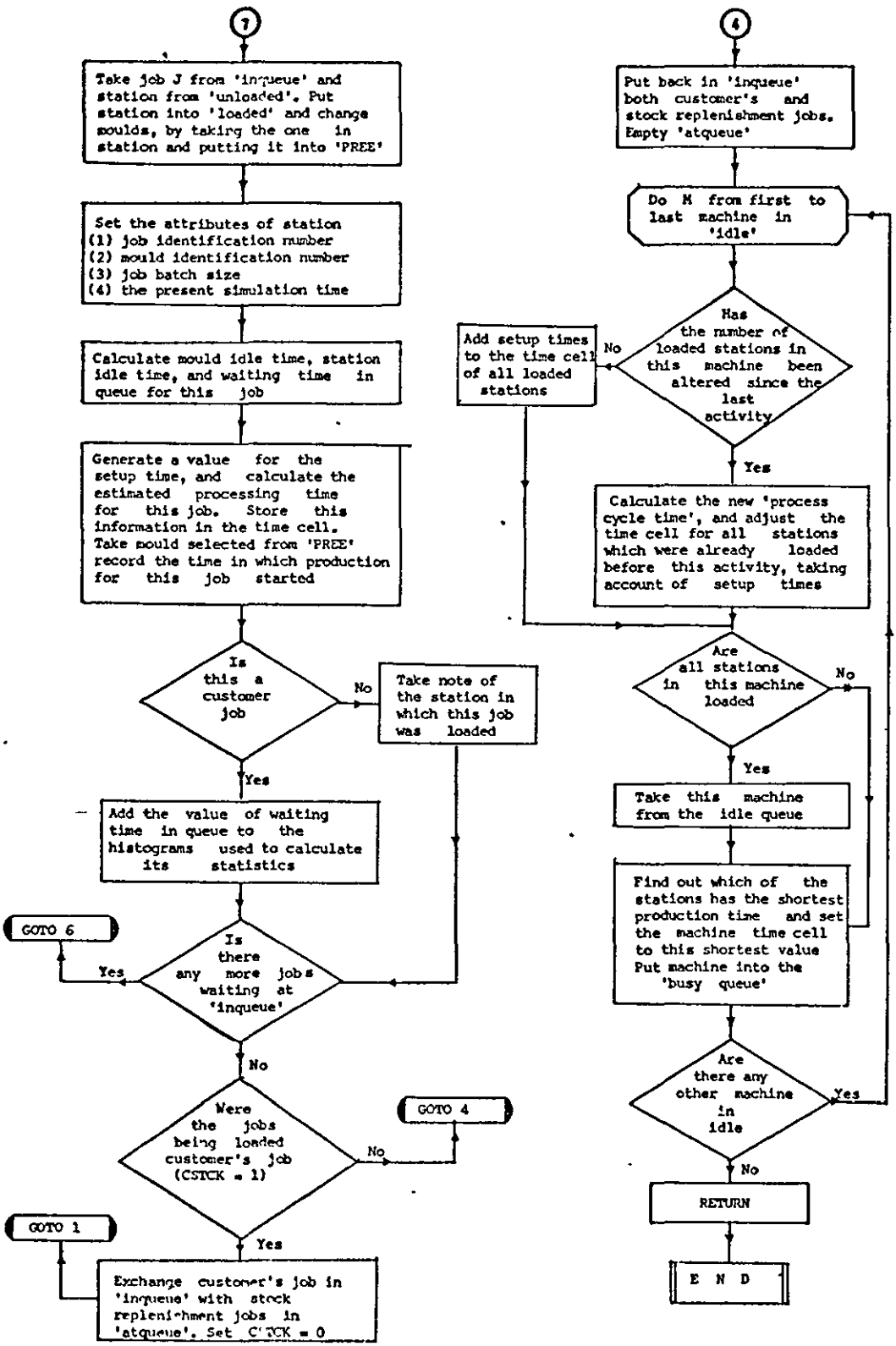


FIGURE A1 - 4

FLOW CHART OF SUBROUTINE BLIC FOR LOADING JOBS INTO MACHINES

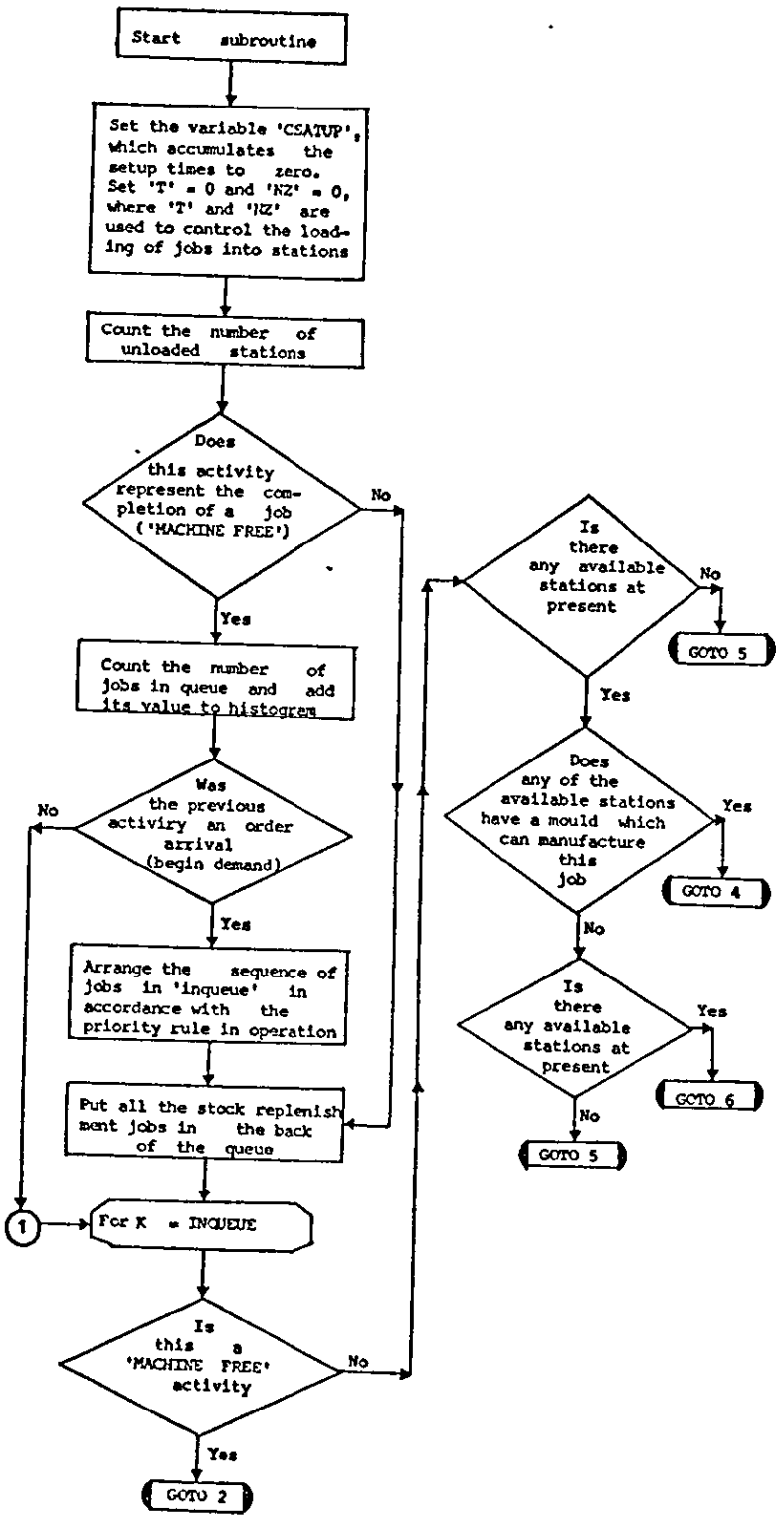


FIGURE A1 - 4 (CONTINUATION)

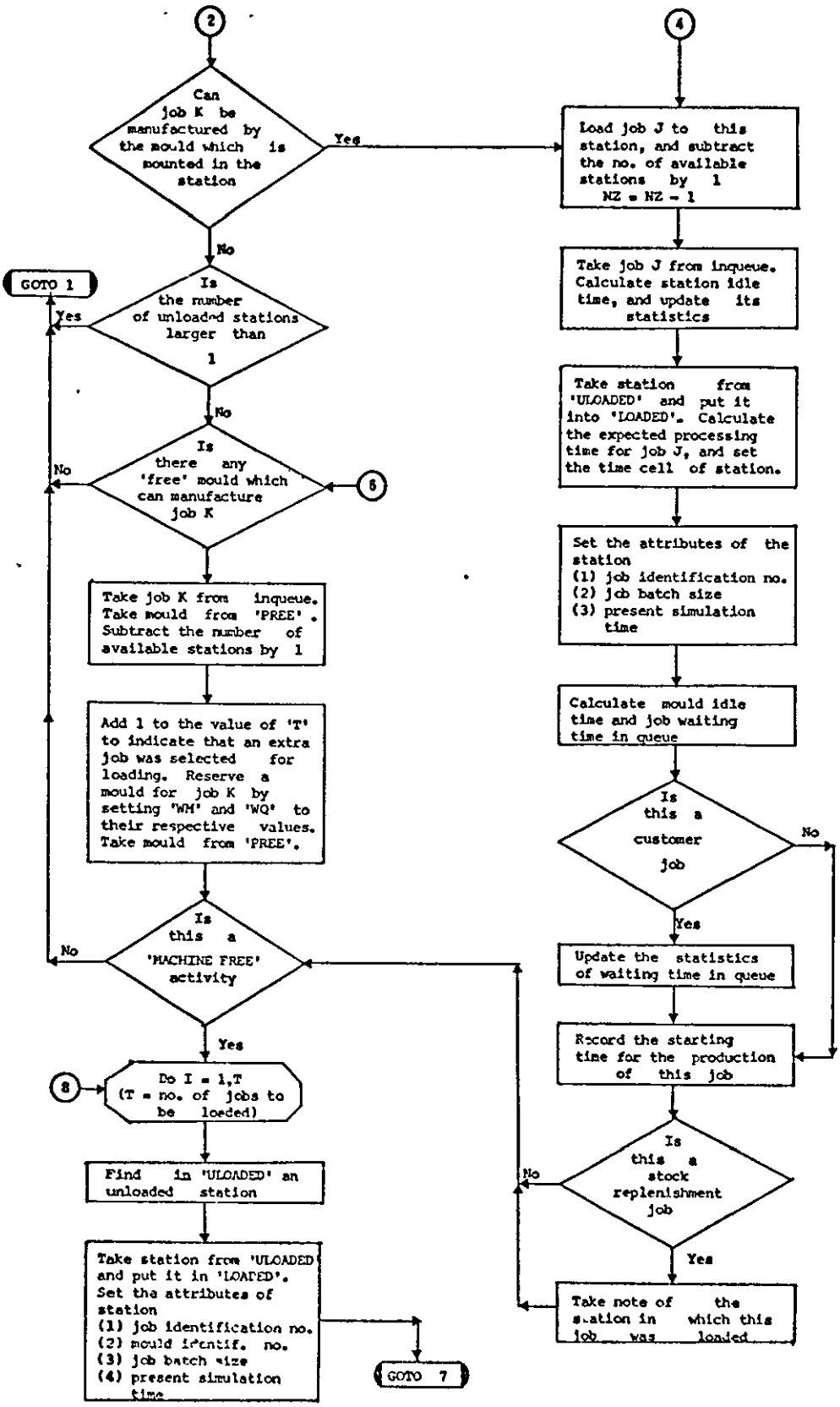


FIGURE A1 - 4 (CONTINUATION)

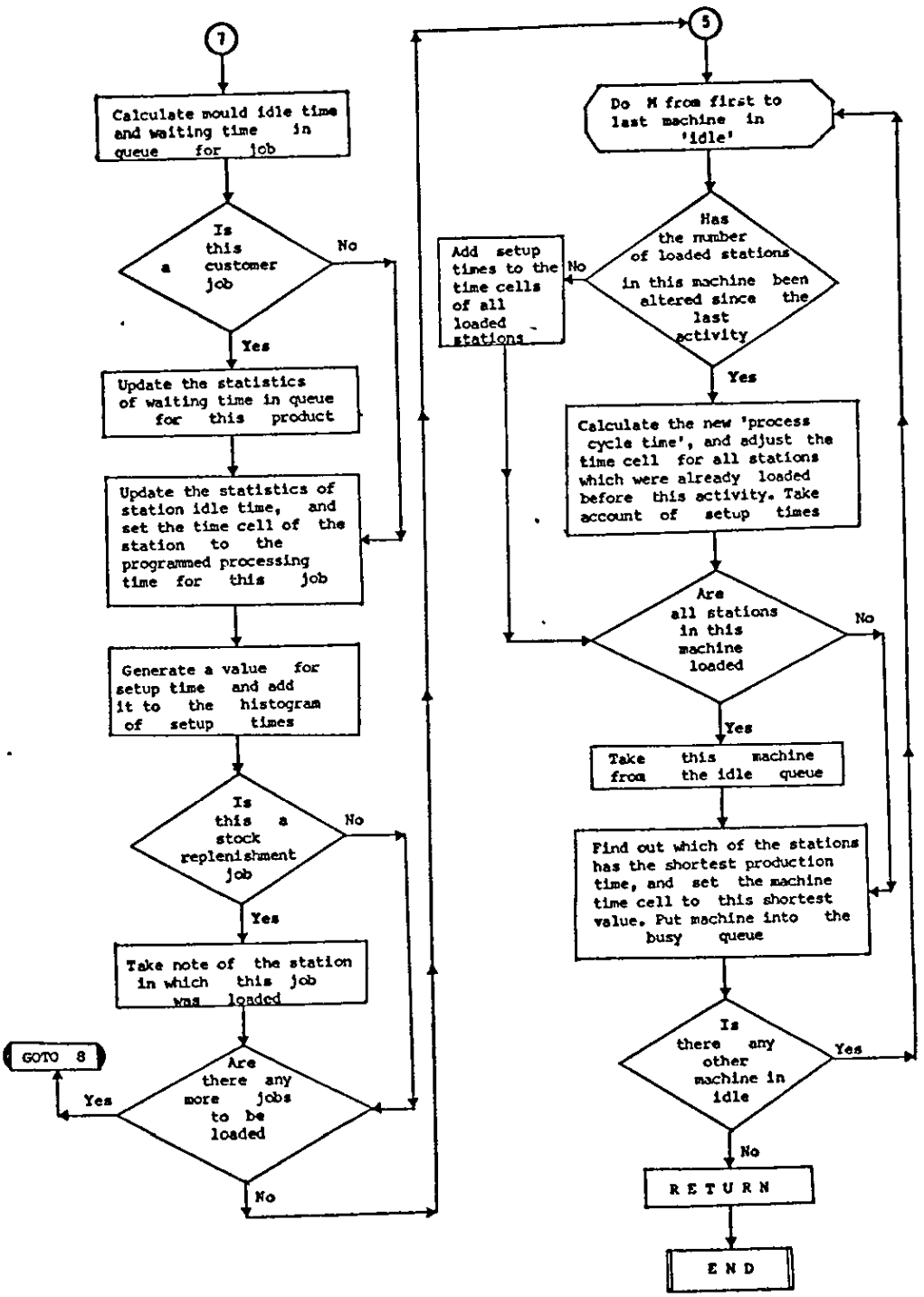
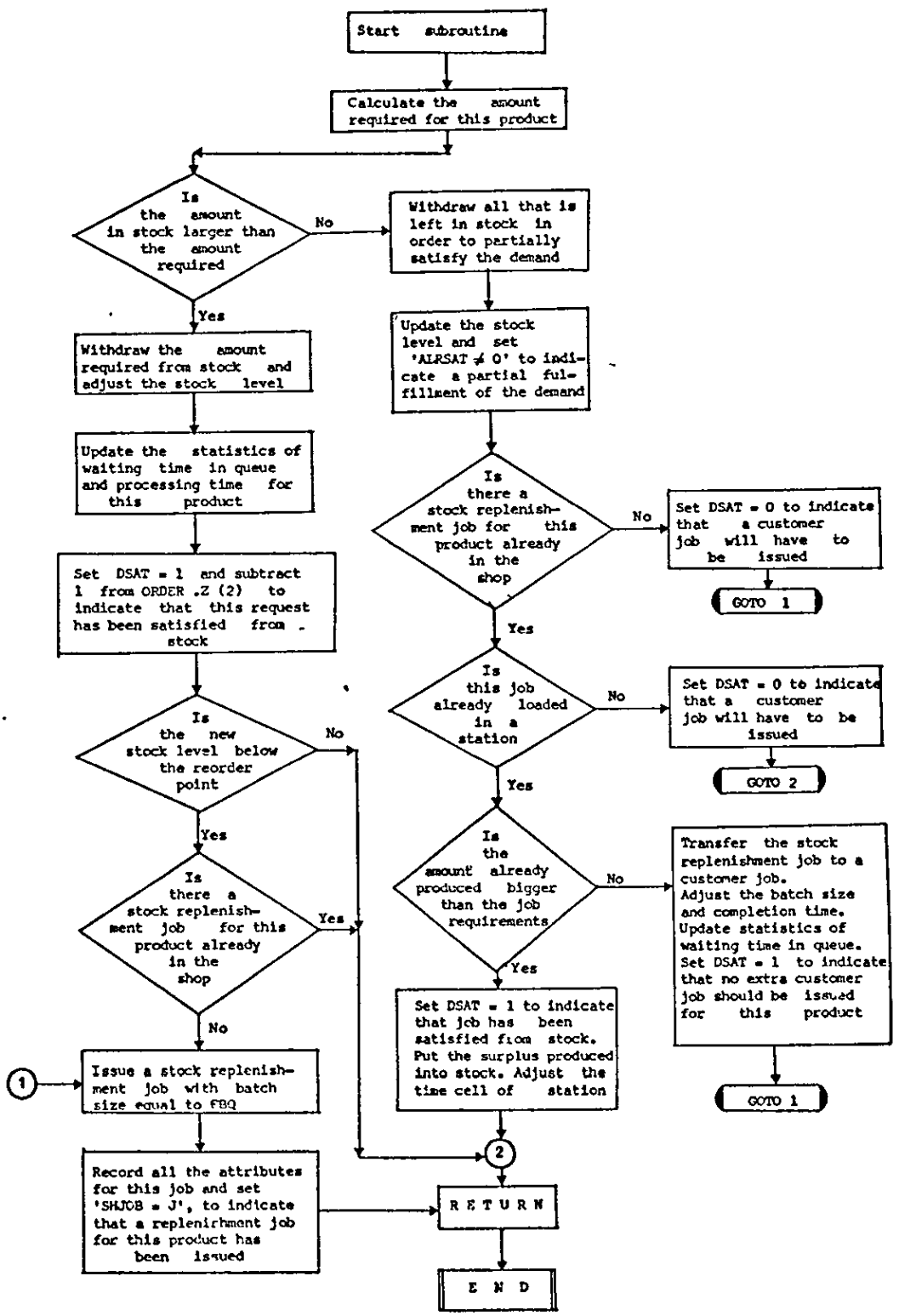


FIGURE A 1 - 5

FLOW CHART OF SUBROUTINE STCT

(STOCK CONTROL)



A1 - 5 Listing of the main program and subroutine

The main program and all subroutines are listed in the following pages. This listing is a straight copy of the job (deck of cards) which was dumped in binary version into the filing system.

```
JOB PFF3,J,PFF 1751
VOLUME 7500
JOBCORE 38K
EDIT $EN/PROGDATA
SELECT APPLICATIONS
UP 3
CSLTRAN ,SPL
CSLCOMP ,,SPL,,1
SELECT
USE CR,$EN/PROGDATA
CREATE OUT
USE CP,OUT
RUN CR15(DUMMY),,1000
FDUMP $EN/PFF1BY
GIVE OUT,$EN/PROGOUT
NFLIST $EN/PROGOUT
****
```

```
DOC EDIT
E 3 1;3
  623    1    1    0    0
      0   10004000000      18       1      20       6
      5    2  520 9450    3  217    1   45
E 2 6;7
   42.0   42.0   42.0    0.0    0.0    0.0 1600.0 1600.0 1600.0    0.0
    0.0    0.0    8.0    1.5    0.0    0.0    0.0    0.0    0.0    0.0
E 3 28;30
    0    0    0    1    1    1    1    1    1    1    1    0    0    0
    2    2   18   22   48   54   68   56   60   32   30   12    8    0
    6    6   54   66  144  162  204  168  180   96   90   36   24    0
E 3 35;37
    0    0    0    0    1    1    1    1    1    1    1    0    0    0
    2    2   18   22   48   54   68   56   60   32   30   12    8    0
    6    6   54   66  144  162  204  168  180   96   90   36   24    0
E 3 42;44
    0    0    0    1    1    1    1    1    1    1    1    0    0    0
    2    2   18   22   48   54   68   56   60   32   30   12    8    0
    6    6   54   66  144  162  204  168  180   96   90   36   24    0
E 7 48;54
   36   37    0   38   39    0   40   41    0   42   43    0    0    0



E 1 62;62
    3    3   19   23   49   55   69   57   61   33   31   13    9    0
E 1 65;65
    3    3   19   23   49   55   69   57   61   33   31   13    9    0
E 1 68;68
    3    3   19   23   49   55   69   57   61   33   31   13    9    0
E 1 73;73
    1    0    0    0
S
****
```

```
DOC CSLCOMP
      SHORTLIST
      SEND TO(ED,SEMICOMPUSER,AXXX)
      DUMP ON(ED,PROGRAM USER)
      WORK(ED,WORKFILEUSER)
      RUN
      LIBRARY(ED,SUBGROUPSRC5)
      LIBRARY(ED,SUBGROUPSRC5)
      PROGRAM(U006)
      MIXED SEGMENTS
      TRACE 1
      INPUT 1,5=CR0
      OUTPUT 2,6=LP0
      OUTPUT 3,9=CP0
      COMPRESS INTEGER AND LOGICAL
      END
```

```
      READ FROM(ED,POSTFILEUSER.AXXX)
      LIST
      FUNCTION    FADF(X,NSTR)
      DIMENSION ISTR(10)
      COMMON/MYR/IND,IP,ISTR
      IF(IND-1)1,2,1
    2 IND=0
        FADF=U*COS(V)
      IF(IP.GE.0)GOTO 3
    4   FADF=  FADF*(-1.0)
    3 CONTINUE
      RETURN
    1 IND=1
      CALL MYRAN(P,NSTR)
      IF(IP.LT.0)GOTO 5
      P=1.0-P
    5 U=SQRT(-2.0*ALOG(P))
      CALL MYRAN(P,NSTR)
      IF(IP.LT.0)GOTO 6
      P=1.0-P
    6 V=6.283185307179 59*P
      FADF=U*SIN(V)
      IF(IP.LT.0)GOTO 4
      RETURN
      END
```

```
FUNCTION    FAEF(AM,S,NSTR)
  FAEF=S*   FADF(X,NSTR)+AM
RETURN
END
```

```
      SUBROUTINE  MYRAN(P,NSTR)
      DIMENSION ISTR(10)
      COMMON/MYR/IND,IP,ISTR
      P= IRAN1(ISTR(NSTR),100001)
      P= (P-0.99)/100000.1
      IF(IP.GE.0)GOTO 100
      P=1.0-P
  100 CONTINUE
      RETURN
      END
```

```
FUNCTION    FABF(A,B,NSTR)
CALL MYRAN(P,NSTR)
  FABF=(B-A)*P+A
RETURN
END
```

```
FUNCTION    FACF(A,NSTR)
CALL MYRAN(P,NSTR)
  FACF=-A*ALOG(P)
RETURN
END
```

```
FUNCTION    FAFF(A,B,NSTR)
FAFF=EXP(FAEF(A,B,NSTR))
RETURN
END
```

```
      SUBROUTINE   FARF(IA,IB,ID,CDF,IREF,IBA1,ID1)
      DIMENSION CDF(IBA1),IREF(ID1)
      CDF(IB)=1
      Y=0.0
      X=1.0/FLOAT(ID)
      IREF(1)=ID
      J=IA
      IDONE=ID+1
      DO 1 I=2,IDONE
      J=J-1
    2 J=J+1
      IF(CDF(J)-Y)2,2,3
    3 IREF(I)=J
      Y=Y+X
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE   FATF(IA,IB,ID,CDF,IREF,JI1,ID1)
      DIMENSION CDF(JI1),IREF(ID1)
      IR=IB-IA+1
      X=1.0/FLOAT(IR)
      Y=0.0
      DO 1 I=IA,IB
      Y=Y+X
      CDF(I+1)=Y
    1 CONTINUE
      IAONE=IA+1
      IBONE=IB+1
      CALL   FARF(IAONE,IBONE,ID,CDF,IREF,JI1,ID1)
      RETURN
      END
```

```
      SUBROUTINE   FAVF(RM,IZ,ID,CDF,IREF,IZ1,ID1)
      DIMENSION CDF(IZ1),IREF(ID1)
      E=EXP(1.0)
      F=RM/E
      S=1.0
      K=RM
      R=K
      IF (K) 8,8,9
    9 DO 1 I=1,K
      P=I
      S=S*F/P
    1 CONTINUE
    8 IF(RM-R)2,3,2
    2 S=S/EXP(RM-R)
    3 CDF(K+1)=S
      IF (K) 5,5,10
   10 I=K-1
    4 W=I+1
      S=S*W/RM
      CDF(I+1)=S
      I=I-1
      IF(I)5,4,4
    5 S=CDF(K+1)
      K1=K+1
      DO 6 I=K1,IZ
      W=I
      CDF(I+1)=S*RM/W
      S=S*RM/W
    6 CONTINUE
      S=CDF(1)
      IZP1=IZ+1
      DO 7 I=2,IZP1
      S=S+CDF(I)
      CDF(I)=S
    7 CONTINUE
      IZONE=IZ+1
      CALL   FARF(1,IZONE,ID,CDF,IREF,IZ1,ID1)
      RETURN
      END
```

```
      SUBROUTINE   FAWF(IA,IB,ID,PDF,IREF,IAB1,ID1)
      DIMENSION PDF(IAB1),IREF(ID1)
      X=PDF(IA)
      IA1=IA+1
      DO 1 I=IA1,IB
      PDF(I)=X+PDF(I)
    1 X=PDF(I)
      CALL   FARF(IA,IB,ID,PDF,IREF,IAB1,ID1)
      RETURN
      END
```

```
      INTEGER FUNCTION   FAZF(N,CDF,IREF,N1,NSTR)
      DIMENSION CDF(N),IREF(N1)
      ID=IREF(1)
      CALL MYRAN(P,NSTR)
      R=P
      JJJ=R*FLOAT(ID)
      M=IREF(JJJ+2)-1
    1 M=M+1
      IF(R-CDF(M))2,1,1
    2   FAZF=M-1
      RETURN
      END
```

```
      FINISH
****
DOC CSLTRAN
      LISTING(3)
      SOURCE(CR,S1)
      OBJECT(ED,POSTFILEUSER,AXXX)
```

```
      MASTER MACH
C
      CLASS ORDER,300(3) SET CUSTQUEUE BSET POOLB
      CLASS JOB,900(4) SET INQUEUE,ATQUEUE BSET POOL
      CLASS TIME MACHINE,2(2) BSET IDLE,BUSY
      CLASS TIME STATION,24(5) BSET LOADED,2,ULOADED,2
      CLASS TIME STYLE,3(4) BSET POOLA
      CLASS MOULD,45(3) BSET PREE
C
      ARRAY STOCK(3,15),CUSTARA(2,9),CUSTARB(2,9),CUSTARC(2,9),VOLDA(2,9
     1),VOLDB(2,9),VOLDC(2,9),MACSET(2,7),STYLA(3,15),IPARM(20),PARM(20)
      ARRAY RSTYLA(3,3,15),TDISA(2,4),TDISB(2,4),TDISC(2,4),WJOB(45,3),
     1CDEMAND(45),TIDLEST(2),TIDLEMD(45),CTIDLEMD(45),FACHINE(2)
     2,TSTCKST(3),TIMSTCK(3),AVSTCK(3)
      ARRAY CSATUP(2),RVOL(3,3),PRODT(45,2),SSTYLA(3,15),KCD(45)
     1,RDELAY(45)  ,BCMACHINE(2),TJOB(400),QTSTCK(3,15),RPOINT(3,15),
     2ERQ(3,15),SHJOB(3,15),MCJOB(3,15),TORDER(60),STR(10)
C
F     DIMENSION Z(20),UNICUM(11),IUNIINT(11),POICUM(31),IPOIINT(31),
     1PROCUM(14),IPROINT(14),N(20),CMD(135,20),IGENSTR(30)
C
      HIST DELAY(40,0,1)  ,WAITQUE,2(120,120,10),SETIDLE(45,0,1),
     1NJOBQ(60,2,3)
C
      FLOAT STYLA,AVERA,SDV,RSDV,CDEMAND,LFACT,ACPROD,ACSXQ,RSTYLA
     1,RAVERA,TIDLEST,TIDLEMD,CTIDLEMD,FACHINE,CYCLE,RVOL,SOMXQ,WJOB
      FLOAT PRODT,T,KCD,RDELAY,ARATE,VRATE,BCMACHINE,AVSTCK,TARD,RD
     1,PARM
C
      COMMON/ALLOCATE/SETIDLE,    MACSET,MOULD,PREE,CUSTQUEUE,
     1POOLB,TIDLEST,IDLE,BUSY,ULOADED,ORDERPLAST,CSATUP,CYCLE,VCONV,
     INJOBQ,STARPOINT,TIDLEMD,CTIDLEMD
C
      COMMON/ALLOCATA/OPMACH,IPARM,PARM,EMPIRIC
C
      COMMON/ALLSTCK/JOB,STATION,ORDER,MACHINE,POOL,LOADED,TORDER,TJOB,
     1BCMACHINE,FACHINE,CLOCK,T.MACHINE,T.STATION,T.STYLE,T.CHANGE,
     2MCJOB,WJOB
C
      COMMON/STOCKCONTA/QTSTCK,RPOINT,ERQ,SHJOB,      STYLA,PRODT,
     1CDEMAND,STYLE,TSTCKST,TIMSTCK,AVSTCK
C
F     COMMON/MYR/IND,IP,I060
C
F     INTEGER FAZF
C
      READ(1,10)NUEXP,PRIOR,BOF,SOF,SLACK,EMPIRIC
      READ(1,19)STATISTIC,TSIMUL,STARPOINT,NMOULD,NRUN,SZRUN,TIRUN
      READ(1,10)NDAYS,NMACHB,VCONV,MAXLOT,NSTYL,GENSTR,NIRUN,MXMOULD
      READ(1,20)IPARM,PARM
```

```
      READ(1,11)CUSTARA
      READ(1,11)CUSTARB
      READ(1,11)CUSTARC
      READ(1,11)VOLDA
      READ(1,11)VOLDB
      READ(1,11)VOLDC
      READ(1,11)MACSET
      READ(1,11)TDISA
      READ(1,11)TDISB
      READ(1,11)TDISC
      FOR I=1,NSTYL
        READ(1,15)((RSTYLA(I,J,K),K=1,14),J=1,3)
        READ(1,11) (SSTYLA(I,K),K=1,14)
        READ(1,11)(STOCK(I,K),K=1,14)
        READ(1,11)(RPOINT(I,K),K=1,14)
        READ(1,11)(EBQ(I,K),K=1,14)
      READ(1,16)((MOULD,I(J),J=1,3),I=1,MXMOULD)
      READ(1,15)(BCMACHINE(I),I=1,NMACHB)
      READ(1,15)(FACHINE(I),I=1,NMACHB)
      READ(1,15)RVOL,ARATE,VRATE
F     READ(1,15)PROCUM
F     DO 8004 I103=1,30
F     I100=I100+1
F     IGENSTR(I103)=I100*2+1
F8004 CONTINUE
      WRITE(6,18)NUEXP
      WRITE(6,19)STATISTIC,TSIMUL,STARPOINT,NMOULD,NRUN,SZRUN,TIRUN
      WRITE(6,10)NDAYS,NMACHB,VCONV,MAXLOT,NSTYL,GENSTR,NIRUN,MXMOULD
C
      DIST CUSTARA,CUSTARB,CUSTARC,VOLDA,VOLDB,VOLDC,MACSET
     1,TDISA,TDISB,TDISC
C
F     CALL FAUF(1,06,6,PROCUM,IPROINT,13,13)
F     IP=1
F     ISTOGEN=I100
F     ICRG=0
 8000 DUMMY
F     DO 8003 I103=1,10
F     IJX=I103+ICRG
F     I060(I103)=IGENSTR(IJX)
F8003 CONTINUE
 8001 DUMMY
      LOAD POOLA,POOLB,POOL,IDLE
C
      WRITE(6,19)GENSTR,PERIOD
      WRITE(6,19)STR
      TSIMUL=TSIMUL*1.0/520.*VCONV
F     IND=0
      READ(1,12)PERIOD,CYCLE
      FOR I=1,NSTYL
```

```
      READ(1,12)T.STYLE.I,(STYLA(I,J),J=1,13)
      READ(1,14)(STYLA(I,K),K=14,15),(STYLE.I(Z),Z=1,4)
      READ(1,11)(QTSTCK(I,K),K=1,14)
C
      Z=NMACHB*12
C
      READ(1,13)((STATION.I(J),J=1,2),I=1,Z)
      READ(1,16)((MACHINE.M(I),I=1,2),M=1,NMACHB)
      NIRUN EQ 1 @ 8002
      WRITE(6,20)IPARM,PARM
      WRITE(6,12)PERIOD,CYCLE
      WRITE(6,19)STR
      WRITE(6,13)CUSTARA,CUSTARB,CUSTARC,VOLDA,VOLDB,VOLDC,MACSET
     1,TDISA,TDISB,TDISC
      FOR I=1,NSTYL
        WRITE(6,12) T.STYLE.I,(STYLA(I,J),J=1,13)
        WRITE(6,14)(STYLA(I,K),K=14,15),(STYLE.I(J),J=1,4)
        WRITE(6,15)((RSTYLA(I,J,K),K=1,14),J=1,3)
        WRITE(6,11)(SSTYLA(I,K),K=1,14)
        WRITE(6,11)(STOCK(I,K),K=1,14)
        WRITE(6,11)(QTSTCK(I,K),K=1,14)
        WRITE(6,11)(RPOINT(I,K),K=1,14)
        WRITE(6,11)(EBQ(I,K),K=1,14)
      WRITE(6,13)((STATION.I(J),J=1,2),I=1,Z)
      WRITE(6,16)((MOULD.I(J),J=1,3),I=1,MXMOULD)
      WRITE(6,16)((MACHINE.M(I),I=1,2),M=1,NMACHB)
      WRITE(6,14)(RCMACHINE(I),I=1,NMACHB)
      WRITE(6,15)(FACHINE(I),I=1,NMACHB)
      WRITE(6,15)RVOL,ARATE,VRATE
F     WRITE(6,15)PROCUM
C
   10 FORMAT(9I5,F5.2,4I5)
   11 FORMAT(14I5)
   12 FORMAT(I5,13F5.2)
   13 FORMAT(14I5)
   14 FORMAT(2F5.2,4I5)
   15 FORMAT(14F5.3)
   16 FORMAT(14I5)
   18 FORMAT(//,45X,11H EXP NUMBER,I5,//)
   19 FORMAT(10I7)
   20 FORMAT(10I5,/,10I5,/,10F7.1,/,10F7.1)
C
 8002 DUMMY
      COMORD=0
      ALRSAT=0
      FOR I=1,NMACHB
        CSATUP(I)=0
        TIDLEST(I)=0
      FOR I=1,NSTYL
        TSTCKST(I)=0
```

```
          AVSTCK(I)=0
          TIMSTCK(I)=0
          FOR J=1,15
            MCJOB(I,J)=0
            SHJOB(I,J)=0
            STOCK(I,J) NE 0 @ 24
            TSTCKST(I)+QTSTCK(I,J)
   24       DUMMY
        T.CHANGE=2400
        CHAVE=0
        FOR I=1,NMACHB
         K=12*I
         J=12*(I-1)+1
         FOR L=J,K
          STATION.L TO ULOADED.I
         DUMMY
        FOR I=1,NMACHB
         MACHINE.I(1) EQ 0 @ 21
         FOR J=ULOADED.I
          MOULD.(STATION.J(2))(1) NE 0 @ 22
          MOULD.(STATION.J(2)) TO PREE
   22     DUMMY
         MACHINE.I FROM IDLE
   21    DUMMY
   17   DUMMY
        K=15*NSTYL
        FOR I=1,K
          CDEMAND(I)=0
          TIDLEMD(I)=0
          CTIDLEMD(I)=0
          FOR J=1,NSTYL
           WJOB(I,J)=0
           RDELAY(I)=0
          FOR J=1,2
           PRODT(I,J)=0
        DUMMY
        J=NMACHB*12+1
        FOR I=J,NMOULD
         MOULD.I TO PREE
        OPMACH=0
        FOR I=1,NMACHB
         MACHINE.I(1) NE 0 @ 23
         OPMACH+1
   23    DUMMY
        DUMMY
C
C
C
C
C
```

```
      ACTIVITIES
C
      BEGIN CHANGE OF PERIOD
C
      COMORD GE SZRUN @ 100
      COMORD=0
      PERIOD + 1
F     KF=I093*I101
F     LF=KF-I093+I106
C     WRITE(6,120)PERIOD,CLOCK
C 120 FORMAT(8H PERIOD=,I3,7H CLOCK=,I7)
F     CMD(44,LF)=I000
      FOR I=1,1
        ACPROD=0
        CXA=0
        ACSXQ=0
        FOR J=1,177,3
          XA=YIELD(NJOBQ,J)
          PROD=XA*J
          ACPROD+PROD
          SOMXQ=XA*(J**2)
          ACSXQ+SOMXQ
          CXA+XA
        AVERA=ACPROD/(CXA*1.0)
        SDV=(ACSXQ-((ACPROD**2)/(CXA*1.0)))/(CXA-1)
        SDV=SDV**0.5
        RSDV=SDV/AVERA
C       WRITE(6,4103)AVERA,SDV,RSDV
C4103   FORMAT(  26H AVER NO OF JOBS IN QUEUE=,F5.1,3X,4H SD=,F5.1,3X,5H
C     1 RSD=,F5.1  )
F     CMD(1,LF)=A065
      T.CHANGE=VCONV*NDAYS
C
      CYCLE=0
      FOR I=1,OPMACH
        ACPROD=0
        CXA=0
        ACSXQ=0
        FOR J=125,1300,10
          XA=YIELD(WAITQUE,I,J)
          PROD=XA*J
          ACPROD+PROD
          SOMXQ=1.0*XA*((1.0*J)**2)
          ACSXQ+SOMXQ
          CXA+XA
        CXA LT 2 @ 4117
        CXA=2
 4117   AVERA=ACPROD/CXA
        SDV=(ACSXQ-((ACPROD**2)/(CXA*1.0)))/(CXA-1)
        SDV LT 0 @ 4124
```

```
        SDV=0
 4174   SDV=SDV**0.5
        AVERA EQ 0 @ 4118
        AVERA=1
 4118   RSDV=SDV/AVERA
        AVERA=AVERA/60.
        CYCLE+AVERA
        SDV=SDV/60.
C       WRITE(6,4102)I,CXA,AVERA,SDV,RSDV
C4102   FORMAT(/,5H MACH,I3,3X,21H NO OF JOBS PROCESSED,I5,3X,16H AVER C
C     1YCLE TIME,F6.2,4H MIN,3X,4H SD=,F6.2,3X,5H RSD=,F6.2,/)
      AVERA=0
      CYCLE=CYCLE/OPMACH
F     CMD(2,LF)=A072
F     CMD(3,LF)=A066
      J=NSTYL*15
      FOR I=1,J
        AVERA+CDEMAND(I)
      LFACT=AVERA*CYCLE/CLOCK/12/OPMACH*100.0
C     WRITE(6,4106)LFACT,AVERA
C4106 FORMAT(/,27H LOAD FACTOR ON THE SYSTEM=,F6.2,3X,18H VOLUME OF DEMA
C     1ND=,F10.1,/)
 4133 FORMAT(F6.2,F10.1)
F     CMD(4,LF)=A068
F     CMD(5,LF)=A065
C
      STARPOINT EQ 4000000 @ 4155
      FOR I=1,J
       WJOB(I,2) LT 2  @ 4125
       CDEMAND(I)=0
       TIDLEMD(I)=0
       GOTO 4128
 4125  CDEMAND(I)=WJOB(I,1)/WJOB(I,2)/VCONV
       TIDLEMD(I)=(WJOB(I,3)-((WJOB(I,1)**2)/(WJOB(I,2))))/(WJOB(I,2)-1)
       CDEMAND(I) LT 0.10 @ 4146
       TIDLEMD(I)=0
       GOTO 4128
 4146  TIDLEMD(I)=(TIDLEMD(I)**0.5)/CDEMAND(I)/VCONV
 4128  DUMMY
F      CMD(I103+45,LF)=A038(I103)
F      CMD(I103+90,LF)=A040(I103)
C     WRITE(6,4119) CDEMAND
C     WRITE(6,4119) TIDLEMD
 4155 DUMMY
      ACPROD=0
      CXA=0
      FOR J=0,44,1
        XA=YIELD(SETIDLE,J)
        PROD=XA*J
        ACPROD+PROD
```

```
      AVERA=ACPROD*100.0/(CLOCK*OPMACH)
C     WRITE(6,4105)AVERA
C4105 FORMAT(/,40H PERC OF MACHINE IDLE TIME DUE TO SETUP=,F6.2,/)
F     CMD(6,LF)=A065
      FOR I=1,OPMACH
        RAVERA=TIDLEST(I)*100.0/CLOCK/12.0
C       WRITE(6,4108)I,RAVERA
      DUMMY
C4108 FORMAT(/,5H MACH,I2,2X,71H PERC OF TOTAL AVAILABLE TIME THAT STATI
C    1ON IS IDLE DUE TO LACK OF JOBS=,F6.2,/)
F     CMD(7,LF)=A071
      FOR I=1,1
        ACPROD=0
        CXA=0
        ACSXQ=0
        FOR J=0,39,1
          XA=YIELD(DELAY,J)
          PROD=XA*J
          ACPROD+PROD
          SQMXQ=1.0*XA*((1.0*J)**2)
          ACSXQ+SQMXQ
          CXA+XA
          KCD(J+1)=CXA
        CXA LT 2 @ 4114
        CXA=2
 4114   AVERA=ACPROD/(CXA+1.0)
        SDV=(ACSXQ-((ACPROD**2)/(CXA+1.0)))/(CXA-1)
        SDV=SDV**0.5
        AVERA EQ 0 @ 4147
        AVERA =0.5
 4147   RSDV=SDV/AVERA
C       WRITE(6,4101)I,AVERA,SDV,RSDV,CXA
C4101   FORMAT(/,5H PROD,I3,3X,12H AVER DELAY=,F6.2,5H DAYS,3X,4H SD=,
C    1F6.2,3X,6H RSDV=,F6.2,3X,14H NO OF ORDERS=,I4,/)
F       CMD(8,LF)=A065
F       CMD(9,LF)=A066
        K=10
F       IC=0
 4150   TARD=0
        SDV=0
        FOR J=K,39
         TARD=TARD+((KCD(J)-KCD(J-1))/CXA*(J-K+1))
        SDV=(1.0-(KCD(K-1)/CXA))*100.0
C       WRITE(6,4149)TARD,SDV
C4149   FORMAT(/,12H TARD INDEX=,F6.2,10X,11H PERC LATE=,F6.2)
F       IC=IC+1
F       CMD(IC+14,LF)=A077
F       CMD(IC+21,LF)=A066
        K+2
        K GT 22 @ 4150
```

```
C
      FOR I=1,1
        ACPROD=0
        CXA=0
        ACSXQ=0
        FOR J=1,45,1
          XA=RDELAY(J)
          ACPROD+(XA*1.0*(J-1))
          CXA+XA
          KCD(J)=CXA
        AVERA=ACPROD/(CX4+1.0)
C       WRITE(6,4141)I,AVERA,CXA
C4141   FORMAT(/,5H PROD,I3,10X,16H REL AVER DELAY=,F6.2,10X,8H DEMAND=,
C    118,/)
F       CMD(10,LF)=A065
F       CMD(11,LF)=I113
        K=10
F       IC=0
 4152   TARD=0
        SDV=0
        FOR J=K,45
         TARD=TARD+((KCD(J)-KCD(J-1))/CXA*(J-K+1))
        SDV=(1.0-(KCD(K-1)/CXA))*100.0
C       WRITE(6,4149)TARD,SDV
F       IC=IC+1
F       CMD(IC+28,LF)=A077
F       CMD(IC+35,LF)=A066
        K+2
        TARD=0
        K GT 22 9 4152
      FOR I=1,NSTYL
        AVSTCK(I)=AVSTCK(I)/CLOCK
F       CMD(I103+11,LF)=A045(I103)
C       WRITE(6,4148)I,AVSTCK(I)
C4148 FORMAT(/,50X,24H AVERAGE STOCK FOR STYLE,I3,F8.0,/)
      CLEAR NJORQ,DELAY,WAITQUE.1,WAITQUE.2,SETIDLE
      FOR I=1,45
        RDELAY(I)=0
  100 DUMMY
```

```
C
      BEGIN DEMAND
C
      FOR I=POOLA
        T.STYLE.I EQ 0 @ 1000
        FIND.Z POOLB FIRST 1006 @ 1005
1005  WRITE(6,1007)
1007  FORMAT(20H POOLB IS NOT ENOUGH)
        EXIT
1006  TORDER(Z)=CLOCK
        ORDER.Z FROM POOLB
        ORDER.Z TO CUSTQUEUE
        PRISET(4) CUSTQUEUE
        ORDER.Z(1)=I
        ORDER.Z(2)=STYLE.I(1)
        ORDER.Z(3)=STYLE.I(2)
        STYLE.I(1)=0
C
9101    DUMMY
        AVSTCK(I)+((1.0*TSTCKST(I))*(1.0*(CLOCK-TIMSTCK(I))))
        TIMSTCK(I)=CLOCK
        FOR K=STYLE.I(3),STYLE.I(4)
         J=K+15*(I-1)
         STYLA(I,K) EQ 0 1009 @ 0
         CDEMAND(J)+(STYLE.I(2)*STYLA(I,K)+3)
         STOCK(I,K) NE 0 @ 1015
         CALL ESTCT(I,K,DSAT,ALRSAT,Z,INQUEUE)
         DSAT EQ 0 @ 1009
1015     SSTYLA(I,K) EQ 0 @ 1011
         Y=1
         GOTO 1012
1011     T=(STYLE.I(2)*STYLA(I,K)+3.0)/MAXLOT
         Y=1+T
         STYLA(I,K)=STYLA(I,K)/Y
         ORDER.Z(2)+(Y-1)
1012     FOR X=1,Y
          FIND J POOL FIRST 1003 @ 1001
1001      WRITE(6,1002)
1002      FORMAT(20H POOL IS NOT ENOUGH )
          EXIT
1003      DUMMY
          JOB.J FROM POOL
          JOB.J(1)=Z
          JOB.J(2)=K
          JOB.J(3)=STYLE.I(2)*STYLA(I,JOB.J(2))+3-ALRSAT
          JOB.J(4)=K+15*(I-1)
          ALRSAT=0
C
9130      DUMMY
          JOB.J TO INQUEUE
```

```
          TJOB(J)=CLOCK
 1009     DUMMY
          DUMMY
C
          ORDER.Z(2) EQ 0 @ 1018
          RDELAY(1)+ORDER.Z(3)
          ADD 0,DELAY
          COMORD+1
          ORDER.Z FROM CUSTQUEUE
          PRISET(4) CUSTQUEUE
          ORDER.Z TO POOLB
          CHECK Z,ORDER.Z(2)
 1018     DUMMY
          SOF NE 0 @ 1016
          RANK K INQUEUE(-JOB,K(3))
          GOTO 1017
 1016     SLACK NE 0 @ 1017
          RANK K INQUEUE(JOB,K(3))
          GOTO 1014
 1017     BOF EQ 0 1014 @ 0
          RANK K INQUEUE(JOB,K(3))
 1014     LS=1
          FIND M IDLE FIRST @ 1008
          CSR=1
          PRIOR EQ 1 1013 @ 0
          PRIOR EQ 2 @ 1013
          CALL EBLLC(M,CSR,S,INQUEUE,LS,SOF,SLACK)
          GOTO 1008
 1013     CALL   EALLC(M,CSR,S,INQUEUE,LS,SOF,SLACK)
 1008     ATQUEUE GAINS INQUEUE
          ZERO INQUEUE
          EMPIRIC EQ 1 @ 1019
          T.STYLE.I=SAMPLE(I,CUSTARA,STR(1),CUSTARB,STR(2),CUSTARC,STR(3))
          STYLE.I(2)=SAMPLE(I,VOLDA,STR(4),VOLDB,STR(5),VOLDC,STR(6))
          K=SAMPLE(I,TDISA,STR(8),TDISB,STR(9),TDISC,STR(10))
          GOTO 1020
 1019     DUMMY
F         NSTR=I103
F         J=I103
F         N2(26+I103)=FACF(A032(J),NSTR)
F         IF(N2(26+I103).EQ.0)N2(26+I103)=1
F         NSTR=I103+3
F         J=I103+6
F         I017(I103,2)=FACF(A032(J),NSTR)
F         IF(I017(I103,2).GT.5000)I017(I103,2)=5000
F         IF(I017(I103,2).LT. 50)I017(I103,2)=50
F         NSTR=I103+6
F         I105=FAZF(14,PROCUM,IPROINT,14,NSTR)
 1020     DUMMY
          T.STYLE.I=(T.STYLE.I)*60./520.*VCONV*ARATE
```

```
          STYLE.I(2)=STYLE.I(2)*RVOL(K,I)*VRATE
          FOR Z=STYLE.I(3),STYLE.I(4)
            STYLA(I,Z)=RSTYLA(I,K,Z)
            STYLA(I,Z) EQ 0 1010 @ 0
            STYLE.I(1)+1
 1010       DUMMY
 1000     DUMMY
C
```

```
C
      BEGIN MACHINE FREE
C
      FOR M=BUSY
        T.MACHINE,M EQ 0 @ 2000
 9104   DUMMY
        FIND J LOADED,M FIRST 2002 @ 2001
          T.STATION,J EQ 0
 2001   WRITE(6,2003)
 2003   FORMAT(30H NO STATION HAS FINISHED A JOB)
        EXIT
 2002   MACHINE,M FROM BUSY
        MACHINE,M TO IDLE
        STATION,J FROM LOADED,M
        STATION,J TO ULOADED,M
        CTIDLEMD(STATION,J(2))=CLOCK
        Y=STATION,J(1)
        JOB,Y TO POOL
        JOB,Y(3) LT 1 2012 @ 0
        L=(CLOCK-TJOB(Y))*60./JOB,Y(3)
        ADD L ,WAITQUE,M
 2012   DUMMY
        X=JOB,Y(1)
 9105   DUMMY
        X EQ 0 @ 2008
        I=JOB,Y(4)/15.+0.99
        AVSTCK(I)+((1.0*TSTCKST(I))*(1.0*(CLOCK-TIMSTCK(I))))
        TSTCKST(I)+JOB,Y(3)
        TIMSTCK(I)=CLOCK
        QTSTCK(I,JOB,Y(2))+JOB,Y(3)
        SHJOB(I,JOB,Y(2))=0
        QTSTCK(I,JOB,Y(2)) LT RPOINT(I,JOB,Y(2)) @ 2009
        FIND K POOL FIRST 2010 @ 0
        WRITE(6,2011)
 2011   FORMAT(20H POOL IS NOT ENOUGH )
        EXIT
 2010   JOB,K FROM POOL
        JOB,K(1)=JOB,Y(1)
        JOB,K(2)=JOB,Y(2)
        JOB,K(3)=EBO(I,JOB,Y(2))
        JOB,K(4)=JOB,Y(4)
        CHECK JOB,K(1),K
        JOB,K TO ATQUEUE
        TJOB(K)=CLOCK
        SHJOB(I,JOB,K(2))=K
        MCJOB(I,JOB,K(2))=0
        GOTO 2009
 2008   ORDER,X(2)-1
        ORDER,X(2) EQ 0 @ 2004
        TORDER(X)=(CLOCK-TORDER(X))/VCONV
```

```
        K=TORDER(X)+1
        K GT 45 @ 2007
        K=45
 2007   RDELAY(K)+ORDER.X(3)
        ADD TORDER(X),DELAY
        COMORD+1
        ORDER.X FROM CUSTQUEUE
        PRISET(4) CUSTQUEUE
        ORDER.X TO POOLB
 9106   DUMMY
 2004   DUMMY
        PRODT(JOB.Y(4),1)+(CLOCK-TJOB(Y))
        PRODT(JOB.Y(4),2)+1
 2009   DUMMY
C
 9107   DUMMY
        S=J
        CSR=0
        PRIOR EQ 1 2006 @ 0
        PRIOR EQ 2 @ 2006
        CALL £BLLC(M,CSR,S,ATQUEUE,LS,SOF,SLACK)
        GOTO 2000
 2006   CALL £ALLC(M,CSR,S,ATQUEUE,LS,SOF,SLACK)
C
 2000   DUMMY
C
C
C
```

```
C
C
      CHAVE EQ 0 @ 3101
      COMORD GE STATISTIC @ 3101
      COMORD=0
      CLEAR DELAY,SFTIDLE,NJOBQ,WAITQUE,1,WAITQUE,2
      T.CHANGE=2400
      PERIOD=0
      FOR M=1,NMACHB
       TIDLEST(M)=0
       FOR I=ULOADED,M
        T.STATION.I=0
      K=NSTYL*15
      FOR I=1,K
       CDEMAND(I)=0
       TIDLEMD(I)=0
       CTIDLEMD(I)=0
       FOR J=1,3
        WJOB(I,J)=0
        RDELAY(I)=0
       FOR J=1,2
        PRODT(I,J)=0
      J=NMACHB*12
      FOR I=1,J
       STATION.I(4)=0
      FOR I=1,400
       TJOB(I)-CLOCK
      FOR I=1,60
       TORDER(I)-CLOCK
      FOR I=1,NSTYL
       AVSTCK(I)=0
       TIMSTCK(I)=0
      CLOCK=0
      CHAVE=1
 3101 DUMMY
      PERIOD GE NRUN @ 9000
      FOR M=IDLE
       FOR I=ULOADED,M
        TIDLEST(M)=T.STATION.I
        TIDLEMD(STATION.I(2))+(CLOCK-CTIDLEMD(STATION.I(2)))
      FOR I=PREE
       TIDLEMD(I)+(CLOCK-CTIDLEMD(I))
      GOTO 4001
C
C
C
C     WRITE(9,11)NUEXP,NMACHB
C
      FOR I=1,J
        CDEMAND(I)=CDEMAND(I)*CYCLE*100.00/CLOCK
```

```
      WRITE(6,4115)
 4115 FORMAT(/,45X,39H LOAD FACTOR FOR INDIVIDUAL INSOLE SIZE,//)
      WRITE(6,4119) CDEMAND
C4134 FORMAT(9F8.2,1,6F8.2)
      FOR I=1,J
        CTIDLEMD(I)=TIDLEMD(I)*100.0/CLOCK
        TIDLEMD(I)=100.0
      WRITE(6,4109)
 4109 FORMAT(/,45X,24H PERC OF MOULD IDLE TIME7//)
      WRITE(6,4119) CTIDLEMD
C     WRITE(9,4134)CTIDLEMD
 4119 FORMAT(15F8.2,/)
      FOR I=1,J
        PRODT(I,2) FQ 0 4129 @ 0
        TIDLEMD(I)=PRODT(I,1)/PRODT(I,2)/VCONV
 4129   DUMMY
      WRITE(6,4130)
 4130 FORMAT(/,45X,44H AVERAGE PRODCT TIME FOR INDIV SIZES IN DAYS,/)
      WRITE(6,4119) TIDLEMD
      FOR I=1,J
        TIDLEMD(I)+CDEMAND(I)
      WRITE(6,4131)
 4131 FORMAT(/,45X,38H AVERAGE DELAY AT SHOP FOR INDIV SIZES,/)
      WRITE(6,4119) TIDLEMD
C     WRITE(9,4134)TIDLEMD
 4001 DUMMY
      NIRUN GE TIRUN 9006 @ 0
F     IF(IP.GT.0) GOTO 9004
F     ISTOGEN=I100
F     ICRG=ICRG+10
F     DO 9009 I103=1,10
F     IJX=I103+ICRG
F     I060(I103)=IGENSTR(IJX)
F9009 CONTINUE
F     IP=1
      GOTO 9005
F9004 IP=-1
 9005 NIRUN+1
F     IND=0
      CLOCK=0
      ZERO CUSTQUEUE,POOLB,INQUEUE,ATQUEUE,POOL,BUSY,LOADED.1
      ZERO LOADED.2,ULOADED.1,ULOADED.2,POOLA,FREE
F     IF(IP.GT.0)GOTO 8001
F     I100=ISTOGEN
      GOTO 8000
 9006 DUMMY
      I=VCONV/57
      WRITE(6,4140)NUEXP,PRIOR,BOF,SOF,SLACK,CLOCK ,STATISTIC,NMOULD,
     1I
 4140 FORMAT(//,40X,7H EXP NO7I5,5X,14H PRIORITY RULE,I3,5X,5H BOF=,I3,
```

```
     15X,5H SOF=,I3,5X,7H SLACK=,I3,//,5X,15H LENGTH OF RUN=,I7,5X,
     211H STATISTIC=,I7,5X,10H NO MOULD=,I3,5X,15H HOURS PER DAY=,I3)
F      DO 4156 JF=1,I095,2
F      DO 4156 KF=1,135
F      CMD(KF,JF)=(CMD(KF,JF)+CMD(KF,JF+1))/2.0
F4156 CONTINUE
F      DO 4157 JF=1,I095,2
F      DO 4157 KF=1,135
F      CMD(KF,19)=CMD(KF,19)+CMD(KF,JF)
F      CMD(KF,20)=CMD(KF,20)+CMD(KF,JF)**2
F4157 CONTINUE
F      JF=I095+I093/2
F      LF=JF-1
F      DO 4158 KF=1,135
F      CMD(KF,19)=CMD(KF,19)/JF
F      CMD(KF,20)=(CMD(KF,20)-JF*CMD(KF,19)**2)/LF
F4158 CONTINUE
       WRITE(9,11)NUFXP
F      WRITE(9,4163)((CMD(KF,JF),JF=1,I095,2),KF=1,135)
F4163 FORMAT(6F10.2)
F      WRITE(6,4164)
F4164 FORMAT(//,40X,29H MEAN AND VARIANCE OF RESULTS,/)
F      WRITE(6,4165)((CMD(KF,JF),JF=19,20),KF=1,45)
F4165 FORMAT(30X,2F16.3,/)
       WRITE(6,4120)
 4120 FORMAT(/,45X,48H AVER WAITING TIME AT QUEUE FOR INDIVIDUAL SIZES,/
     1/)
F      WRITE(6,4119)(CMD(KF,19),KF=46,90)
       WRITE(6,4121)
 4121 FORMAT(/,45X,28H RELATIVE STANDARD DEVIATION,//)
F      WRITE(6,4119)(CMD(KF,19),KF=91,135)
F      WRITE(6,4161)
F4161 FORMAT(//,20X,32H RESULTS OF PAIRS OF ANTITHETICS,//)
       EXIT
 9000 DUMMY
       END
```

```
      SUBROUTINE EALLC(M,CSR,S,INQUEUE,LS,SOF,SLACK)
C
C           MOULD PRIORITY RULE
C
      CLASS ORDER,500(3) SET CUSTQUEUE RSET POOLB
      CLASS JOB,900(4) SET INQUEUE,ATQUEUE RSET POOL
      CLASS TIME MACHINE,2(2) BSET IDLE,BUSY
      CLASS TIME STATION,24(5) BSET LOADED,2,ULOADED,2
      CLASS TIME FALS,4(0)
      CLASS MOULD,45(3) BSET PREE
C
      ARRAY MACSET(2,7),N(2),TIDLEST(2),TIDLEMD(45),
     1CTIDLEMD(45),FACHINE(2),WJOB(45,3),CSATUP(2),BCMACHINE(2)
     2,TORDER(500),TJOB(400),MCJOB(3,15),STR(10),IPARM(20);PARM(20)
C
      HIST SETIDLE(45,0,1),NJOBQ(60,2,3)
C
      FLOAT TIDLEST,TIDLEMD,CTIDLEMD,FACHINE,CYCLE,WJOB,BCMACHINE,PARM
C
      COMMON/ALLOCATE/SETIDLE,    MACSET,MOULD,PREE,CUSTQUEUE;-
     1POOLB,TIDLEST,IDLE,BUSY;ULOADED,ORDERPLAST,CSATUP,CYCLE,VCONV,
     INJOBQ,STARPOINT,TIDLEMD,CTIDLEMD
C
      COMMON/ALLOCATA/OPMACH,IPARM;PARM,EMPIRIC
C
      COMMON/ALLSTCK/JOB,STATION,ORDER,MACHINE;POOL,LOADED,TORDER,TJOB,
     1BCMACHINE,FACHINE,CLOCK;T.MACHINE,T.STATION,T.FALS,MCJOB,WJOB
C
F     COMMON/HYR/IND,IP,1039
C
 9121 DUMMY
      FOR I=1,OPMACH
        CSATUP(I)=0
      NZ=0
      FOR I=IDLE
        COUNT Z ULOADED.I
        N(I)=Z
        NZ=NZ+N(I)
      JA=S
      CSR EQ 0 @ 3028
      COUNT K INQUEUE
      ADD K,NJOBQ
 9122 DUMMY
      LS EQ 1 @ 3028
      SOF NE 0 @ 3021
      RANK K INQUEUE(-JOB,K(3))
      GOTO 3028
 3021 DUMMY
      SLACK EQ 0 3028 @ 0
      RANK K INQUEUE ((JOB,K(3)*4.5)+(CLOCK-TJOB(K)))
```

```
3028 SPLIT J INQUEUE INTO ATQUEUE
      JOB.J(1) EQ 0
     INQUEUE LOSES ATQUEUE
     LS=0
     CSTCK=1
3035 FOR K=INQUEUE
       CSR EQ 0 @ 3022
       FOR I=1,3
         MOULD,(STATION,JA(2))(I) EQ JOB;K(4) 3023 @ 0
         DUMMY
       GOTO 3003
3022   DUMMY
       FOR MI =IDLE
         ULOADED,MI EMPTY @ 3002
         DUMMY
       GOTO 3004
3002   DUMMY
       FOR MI=IDLE
         M=MI
         FOR I=1,3
          FIND JA ULOADED,M FIRST 3023 @ 0
           MOULD,(STATION,JA(2))(I) EQ JOB.K(4)
          DUMMY
       GOTO 3003
3023   DUMMY
9111   DUMMY
       JOB,K FROM INQUEUE
       SATUP=1
       CSATUP(M)+SATUP
       ADD SATUP,SETIDLE
       TIDLEST(M)-(T.STATION,JA)
       STATION,JA FROM ULOADED,M
       STATION,JA TO LOADED,M
       COUNT Z LOADED,M
       T.STATION,JA=JOB,K(3)*(BCMACHINE(M)+FACHINE(M)*Z)+0.5
       STATION,JA(3)=JOB,K(3)
       STATION,JA(4)=CLOCK
       STATION,JA(1)=K
       J = STATION,JA(2)
       TJOB(K)=CLOCK-TJOB(K)
       TIDLEMD(J)+(CLOCK-CTIDLEMD(J))
       JOB,K(1) NE 0 @ 3036
       WJOB(JOB,K(4),1)+TJOB(K)
       WJOB(JOB,K(4),2)+1
       WJOB(JOB,K(4),3)+(1.0*TJOB(K))**2
3036   DUMMY
       TJOB(K)=CLOCK
       JOB,K(1) EQ 0 @ 3024
       I=(JOB,K(4)-JOB,K(2))/15.0+1
       MCJOB(I,JOB,K(2))=JA
```

```
 3024    CSR EQ 0 3004 @ 3003
 3003    DUMMY
C
 9123 DUMMY
 3013 DUMMY
      CSR EQ 0 @ 3027
      MZ EQ 1 @ 3010
 3027 FOR J=INQUEUE
        FOR MI=IDLE
          M=MI
          FIND JA ULOADED,M FIRST 3005 @ 0
          DUMMY
        GOTO 3004
 3005   FOR I=1,3
          FIND K PREF FIRST 3018 @ 0
           MOULD,K(I) EQ JOB,J(4)
          DUMMY
        GOTO 3006
 3018   JOB,J FROM INQUEUE
        STATION,JA TO LOADED,M
        STATION,JA FROM ULOADED,M
        MOULD,(STATION,JA(2)) TO PREE
        STATION,JA(1)=J
        STATION,JA(2)=K
        TIDLEMD(K)+(CLOCK-CTIDLEMD(K))
        TJOB(J)=CLOCK-TJOB(J)
        JOB,J(1) NE 0 @ 3037
        WJOB(JOB,J(4),1)+TJOB(J)
        WJOB(JOB,J(4),2)+1
        WJOB(JOB,J(4),3)+(1.0*(TJOB(J)))**2
 3037   DUMMY
        TJOB(J)=CLOCK
        TIDLEST(M)=(T,STATION,JA)
        COUNT Z LOADED,M
        T,STATION,JA=JOB,J(3)*(BCMACHINE(M)+FACHINE(M)*2)+0.5
        STATION,JA(3)=JOB,J(3)
        STATION,JA(4)=CLOCK
        EMPIRIC EQ 1 @ 3039
        SETUP=SAMPLE(1,MACSET,STR(7))
        GOTO 3040
 3039   DUMMY
F       UL=A041(13)+A041(14)*3
F       LL=A041(13)-A041(14)*3
F       I059=FAFF(A041(13),A041(14),10)
F       IF(I059.GT.UL)I059=UL
F       IF(I059.LT.LL)I059=LL
 3040   CSATUP(M)+SETUP
        ADD SETUP,SETIDLE
        MOULD,K FROM PREE
 9117   DUMMY
```

```
       JOB.J(1) EQ 0 @ 3038
       I=(JOB.J(4)-JOB.J(2))/15.0+1
       MCJOB(I,JOB.J(2))=JA
3038   CSR EQ 0 3004 @ 3006
3006   DUMMY
3010 CSTCK EQ 1 @ 3004
     INQUEUE GAINS ATQUEUE
     SPLIT J INQUEUE INTO ATQUEUE
       JOB.J(1) NE 0
     INQUEUE LOSES ATQUEUE
     CSTCK=0
     GOTO 3035
3004 INQUEUE GAINS ATQUEUE
     ZERO ATQUEUE
     FOR M=IDLE
       COUNT Z LOADED.M
       MACHINE.M(2) EQ Z 3030 @ 0
       FOR J=LOADED.M
        CLOCK GT STATION.J(4) @ 3031
        STATION.J(3)=((CLOCK-STATION.J(4))/(BCMACHINE(M)+MACHINE.M(2)
    1+FACHINE(M))+0.5)
3031     T.STATION.J=STATION.J(3)*(BCMACHINE(M)+FACHINE(M)*Z)
    1+CSATUP(M)+0.5
        STATION.J(4)=CLOCK+CSATUP(M)
       MACHINE.M(2)=Z
       GOTO 3032
3030   FOR KA=LOADED.M
        T.STATION.KA+CSATUP(M)
        STATION.KA(4)+CSATUP(M)
3032   ULOADED.M EMPTY @ 3026
       MACHINE.M FROM IDLE
3026   DUMMY
       FIND K LOADED.M MIN(T.STATION.K) @ 3007
       T.STATION.K LE 0 @ 3011
       T.STATION.K =1
3011   DUMMY
       T;MACHINE.M=T.STATION;K
       MACHINE.M TO BUSY
3007   DUMMY
     RETURN
     END
```

```
      SUBROUTINE EBILC(M,CSR,S,INQUEUE,LS,SOF,SLACK)
C
C            STRICT PRIORITY RULE
C
      CLASS ORDER,300(3) SET CUSTQUEUE BSET POOLB
      CLASS JOB,900(4) SET INQUEUE,ATQUEUE BSET POOL
      CLASS TIME MACHINE,2(2) BSET IDLE,BUSY
      CLASS TIME STATION,24(5) BSET LOADED,2,ULOADED,2
      CLASS TIME FALS,4(0)
      CLASS MOULD,45(3) BSET FREE
C
      ARRAY MACSET(2,7),CTIDLEMD(45),WQ(24),WM(24),N(2)
     1,TIDLEST(2),TIDLEMD(45),FACHINE(2) ,WJOB(45,3),CSATUP(2),
     2BCMACHINE(2),TORDER(60),TJOB(400),MCJOB(3,15),STR(10)
C
      ARRAY IPARM(20),PARM(20)
C
      HIST SETIDLE(45,0,1),NJOBQ(60,2,3)
C
      FLOAT TIDLEST,TIDLEMD,CTIDLEMD,FACHINE,CYCLE,WJOB,BCMACHINE,PARM
C
      COMMON/ALLOCATE/SETIDLE,    MACSET,MOULD,FREE,CUSTQUEUE,
     1POOLB,TIDLEST,IDLE,BUSY,ULOADED,ORDERPLAST,CSATUP,CYCLE,VCONV,
     2NJOBQ,STARPOINT,TIDLEMD,CTIDLEMD
C
      COMMON/ALLOCATA/OPMACH,IPARM,PARM,EMPIRIC
C
      COMMON/ALLSTCK/JOB,STATION,ORDER,MACHINE,POOL,LOADED,TORDER,TJOB,
     1BCMACHINE,FACHINE,CLOCK,T.MACHINE,T.STATION,T.FALS,MCJOB,WJOB
C
F     COMMON/MYR/IND,IP,I041
C
 9121 DUMMY
      FOR I=1,OPMACH
        CSATUP(I)=0
      T=0
      NZ=0
      FOR I=IDLE
        COUNT Z ULOADED.I
        N(I)=Z
        NZ=NZ+N(I)
      JA=S
      CSR EQ 0 @ 3029
      COUNT K INQUEUE
      ADD K,NJOBQ
 9122 DUMMY
      LS EQ 1 @ 3035
      SOF NE 0 @ 3028
      RANK K INQUEUE(-JOB,K(3))
      GOTO 3029
```

```
3028 DUMMY
     SLACK EQ 0 3029 @ 0
     RANK P INQUEUE ((JOB.K(3)*4.5)+(CLOCK-TJOB(K)))
3029 DUMMY
3033 SPLIT J INQUEUE INTO ATQUEUE
      JOB J(1) EQ 0
     INQUEUE LOSES ATQUEUE
     INQUEUE GAINS ATQUEUE
     ZERO ATQUEUE
3035 LS=0
     FOR K=INQUEUE
       CSR EQ 0 @ 3026
       JA=S
       FOR I=1,3
         MOULD.(STATION.JA(2))(I) EQ JOB.K(4) 3020 @ 0
         DUMMY
       NZ GT 1 3021 @ 3024
3026   FOR MI=IDLE
         P=MI
         N(MI) EQ 0 3023 @ 0
         FOR I=1,3
          FIND JA ULOADED.M FIRST 3020 @ 0
           MOULD.(STATION.JA(2))(I) EQ JOB.K(4)
          DUMMY
3023     DUMMY
       FOR MI=IDLE
         P=MI
         N(MI) EQ 0 @ 3024
         DUMMY
       GOTO 3022
3024   FOR I=1,3
         FIND J PREE FIRST 3009 @ 0
          MOULD.J(I) EQ JOB.K(4)
         DUMMY
       GOTO 3021
3009   DUMMY
       T=T+1
       WQ(T)=K
       WM(T)=J
       JOB.K FROM INQUEUE
       N(M)=N(M)-1
       NZ-1
       MOULD.J FROM PREE
       GOTO 3027
3020   DUMMY
       N(M)=N(M)-1
       NZ-1
       JOB.K FROM INQUEUE
9111   DUMMY
       SATUP=1
```

```
        CSATUP(M)+SATUP
        ADD SATUP,SFTIDLE
        TIDLEST(M)-(T.STATION.JA)
        STATION.JA FROM ULOADED.M
        STATION.JA TO LOADED.M
        COUNT Z LOADED.M
        T.STATION.JA=JOB.K(3)+(RCMACHINE(M)+FACHINE(M)*Z)+0.5
        STATION.JA(3)=JOB.K(3)
        STATION.JA(4)=CLOCK
        STATION.JA(1)=K
        J = STATION.JA(2)
        TIDLEMD(J)+(CLOCK-CTIDLEMD(J))
        TJOB(K)=CLOCK-TJOB(K)
        JOB.K(1) NE 0 @ 3037
        WJCB(JOB.K(4),1)+TJOB(K)
        WJCB(JOB.K(4),2)+1
        WJCB(JOB.K(4),3)+(1.0*(TJOB(K)))**2
3037    DUMMY
        TJCB(K)=CLOCK
        JOB.K(1) EQ 0 @ 3036
        I=(JOB.K(4)-JOB.K(2))/15.0+1
        MCJOB(I,JOB.K(2))=JA
3036    DUMMY
C
3027    CSR EQ 0 3022 @ 3021
3021    DUMMY
3022 DUMMY
     T EQ 0 3004 @ 0
     FOR I=1,T
       K=WM(I)
       J=WQ(I)
       FOR M=IDLE
         FIND JA ULOADED.M FIRST 3025 @ 0
         DUMMY
       GOTO 3004
3025   STATION.JA FROM ULOADED.M
       STATION.JA TO LOADED.M
       MOULD.(STATION.JA(2)) TO PREE
       STATION.JA(1)=J
       STATION.JA(2)=K
       TIDLEMD(K)+(CLOCK-CTIDLEMD(K))
       TJOB(J)=CLOCK-TJOB(J)
       JOB.J(1) NE 0 @ 3038
       WJCB(JOB.J(4),1)+TJOB(J)
       WJOB(JOB.J(4),2)+1
       WJOB(JOB.J(4),3)+(1.0*(TJOB(J)))**2
3038   DUMMY
       TJOB(J)=CLOCK
       TIDLEST(M)-(T.STATION;JA)
       COUNT Z LOADED.M
```

```
        T.STATION.JA=JOB.J(3)*(BCMACHINE(M)+FACHINE(M)*Z)+0.5
        STATION.JA(3)=JOB.J(3)
        STATION.JA(4)=CLOCK
        EMPIRIC EQ 1 @ 3039
        SETUP=SAMPLE(1,MACSET(STR(7))
        GOTO 3040
 3039   DUMMY
F       UL=A043(13)+A043(14)*3
F       LL=A043(13)-A043(14)*3
F       I061=FAEF(A043(13),A043(14),10)
F       IF(I061.GT.UL)I061=UL
F       IF(I061.LT.LL)I061=LL
 3040   CSATUP(M)+SETUP
        ADD SETUP,SETIDLE
 9117   DUMMY
        JOB.J(1) EQ 0 @ 3006
        L=(JOB.J(4)-JOB.J(2))/15.0+1
        MCJOB(L,JOB.J(2))=JA
 3006   DUMMY
 3004 DUMMY
      FOR M=IDLE
        COUNT Z LOADED.M
        MACHINE.M(2) EQ Z 3030 @ 0
        FOR J=LOADED.M
         CLOCK GT STATION.J(4) @ 3031
         STATION.J(3)-(((CLOCK-STATION.J(4))/(BCMACHINE(M)+MACHINE.M(2)
    1*FACHINE(M))-0.5)
 3031     T.STATION.J=STATION.J(3)*(BCMACHINE(M)+FACHINE(M)*Z)+
     1    CSATUP(M)+0.5
         STATION.J(4)=CLOCK+CSATUP(M)
        MACHINE.M(2)=Z
        GOTO 3032
 3030   FOR KA=LOADED.M
         T.STATION.KA+CSATUP(M)
         STATION.KA(4)+CSATUP(M)
 3032   N(M) EQ 0 @ 3010
        MACHINE.M FROM IDLE
 3010   FIND K LOADED.M MIN(T.STATION.K) @ 3007
        T.STATION.K LE 0 @ 3011
        T.STATION.K=1
 3011   DUMMY
 9124   DUMMY
        T.MACHINE.M=T.STATION.K
        MACHINE.M TO BUSY
 3007   DUMMY
      DUMMY
      RETURN
      END
```

```
      SUBROUTINE ESTCT(I,K,DSAT,ALRSAT,Z,INQUEUE)
C
C       SUBROUTINE FOR STOCK CONTROL
C
      CLASS ORDER,300(3)
      CLASS JOB,900(4) SET INQUEUE BSET POOL
      CLASS TIME MACHINE,2(2)
      CLASS TIME STATION,24(5) BSET LOADED,2
      CLASS TIME STYLE,3(4)
      CLASS TIME FALS,1(0)
C
      ARRAY QTSTCK(3,15),RPOINT(3,15),EBQ(3,15),SHJOB(3,15),MCJOB(3,15)
     1,STYLA(3,15),PRODT(45,2),CDEMAND(45),BCMACHINE(2),FACHINE(2),
     2TJOB(400),TORDER(60),WJOB(45,3),TSTCKST(3),TIMSTCK(3),AVSTCK(3)
C
      FLOAT STYLA,CDEMAND,FACHINE,PRODT,BCMACHINE,WJOB,AVSTCK
C
      COMMON/STCCKCONTA/QTSTCK,RPOINT,EBQ,SHJOB,      STYLA,PRODT,
     1CDEMAND,STYLE,TSTCKST,TIMSTCK,AVSTCK
      COMMON/ALLSTCK/JOB,STATION,ORDER,MACHINE,POOL,LOADED,TORDER,TJOB,
     1BCMACHINE,FACHINE,CLOCK,T.MACHINE,T.STATION,T.STYLE,T.FALS,
     2MCJOB,WJOB
C
      QTJOB=STYLE.I(2)*STYLA(I,K)
      S=K+15*(I-1)
      QTJOB LE QTSTCK(I,K) @ 1001
      QTSTCK(I,K)-QTJOB
      PRODT(S,2)+1
      WJOB(S,2)+1
      TSTCKST(I)-QTJOB
      ORDER.Z(2)-1
      DSAT=1
      QTSTCK(I,K) LE RPOINT(I,K) @ 1000
      SHJOB(I,K) EQ 0 @ 1000
 1009 FIND J POOL FIRST 1004 @ 1002
 1002 WRITE(6,1003)
 1003 FORMAT(20H POOL IS NOT ENOUGH )
      EXIT
 1004 JOB.J FROM POOL
      JOB.J(1)=0
      JOB.J(2)=K
      JOB.J(3)=EBQ(I,K)
      JOB.J(4)=K+15*(I-1)
      JOB.J TO INQUEUE
      TJOB(J)=CLOCK
      SHJOB(I,K)=J
      MCJOB(I,K)=0
      GOTO 1000
 1001 QTJOB-QTSTCK(I,K)
      TSTCKST(I)-QTSTCK(I,K)
```

```
      ALRSAT=QTSTCK(I,K)
      QTSTCK(I,K)=0
      SHJOB(I,K) NE 0 @ 1005
      MCJOB(I,K) NE 0 @ 1006
      JA=MCJOB(I,K)
      M=JA/12.0+0.99
      STATION.JA(3)-(((CLOCK-STATION.JA(4))/(BCMACHINE(M)+MACHINE.M(2)*
     .1FACHINE(M)))
      STATION.JA(4)=CLOCK
      J=SHJOB(I,K)
      QTJOB LE (JOB.J(3)-STATION.JA(3)) @ 1007
      QTSTCK(I,K)+(JOB.J(3)-STATION.JA(3)-QTJOB)
      TSTCKST(I)+(JOB.J(3)-STATION.JA(3)-QTJOB)
      JOB.J(3)=STATION.JA(3)
      TJOB(J)=CLOCK
      T.STATION.JA=STATION.JA(3)+(BCMACHINE(M)+FACHINE(M)*MACHINE.M(2))
      FIND L LOADED.M MIN(T.STATION.L) @ 1008
      T.STATION.L LE 0 @ 1008
      T.STATION.L=1
1008  T.MACHINE.M=T.STATION.L
      ORDER.Z(2)-1
      PRODT(S,2)+1
      WJOB(S,2)+1
      DSAT=1
      ALRSAT=0
      GOTO 1000
1005  DSAT=0
      GOTO 1009
1006  DSAT=0
      GOTO 1000
1007  J=SHJOB(I,K)
      JOB.J(1)=Z
      JOB.J(3)=QTJOB-(JOB.J(3)-STATION.JA(3))
      STATION.JA(3)=JOB.J(3)
      T.STATION.JA=JOB.J(3)+(BCMACHINE(M)+FACHINE(M)+MACHINE.M(2))
      STATION.JA(4)=CLOCK
      WJOB(S,2)+1
      TJOB(J)=CLOCK
      FIND L LOADED.M MIN(T.STATION.L) @ 1010
      T.STATION.L LE 0 @ 1010
      T.STATION.L = 1
1010  T.MACHINE.M=T.STATION.L
      DSAT=1
      ALRSAT=0
      GOTO 1009
1000  DUMMY
      RETURN
      END
```

```
ENDSUBFILE
```

APPENDIX 2

EMPIRICAL DATA USED IN THE PRELIMINARY INVESTIGATION

A2 - 1 Introduction

In this appendix the empirical data used in the preliminary investigation, and the cost data used in the study of strategies for capacity manipulation are presented. It should be pointed out that the distributions of interarrival times and order size, and the distribution of proportions are based on scarce data, as explained in paragraph 4.2. More comprehensive data was obtained later on but was not used in this premilinary investigation.

In the following tables the data used in the preliminary investigation is presented. Tables A2 - 1 to A2 - 3 present the distributions of interarrival times; tables A2 - 4 to A2 - 6 present the distributions of order sizes; tables A2- 7 to A2 - 9 present the distribution of proportions; table A2 - 10 presents the list of moulds, and table A2 - 11 presents the distribution of setup times.

The cost data used in the study of strategies for capacity manipulation is presented in table A2 - 12.

TABLE A2 - 1

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 1

Interarrival times (hours)	Cumulative distribution
18	0.1000
27	0.3000
45	0.4000
81	0.5000
117	0.6000
162	0.7000
171	0.8000
198	1.0000

$\overline{X}$ = 104.4 hours

TABLE A2 - 2

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 2

Interarrival times (hours)	Cumulative distribution
9	0.0714
27	0.2142
45	0.4285
54	0.6428
63	0.7857
72	0.8571
90	0.9285
117	1.0000

$\overline{X}$ = 54.63

TABLE A2 - 3

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 3

Interarrival times (hours)	Cumulative distribution
4	0.3000
9	0.4000
27	0.5000
45	0.6000
63	0.7000
72	0.9000
81	1.0000
90	1.0000

$\bar{X}$ = 36.90 hours

TABLE A2 - 4

DISTRIBUTION OF ORDERS SIZE

PRODUCT STYLE 1

Demand category (class)	Cumulative distribution
500	0.1428
1500	0.2857
2500	0.4286
3500	0.5714
4500	0.7142
5500	0.8571
6500	1.0000
7500	1.0000

Although the mean value of this distribution is 3500, the actual value of the mean order size is 2404. This is because of the correction factors associated with the distribution of proportions (see table A2 - 7 for distribution of proportions).

TABLE A2 - 5

DISTRIBUTION OF ORDER SIZE

PRODUCT STYLE 2

Demand category (class)	Cumulative distribution
500	0.1111
1000	0.2222
1500	0.6667
2000	0.7778
2500	0.7778
3000	0.7778
3500	1.0000
4000	1.0000

Although the mean value of this distribution is 1833, the actual value for mean order size is 1971. This is because of the correction factors associated with the distribution of proportions.(See table A2 - 8 for distribution of proportions).

TABLE A2 - 6

DISTRIBUTION OF ORDER SIZE

PRODUCT STYLE 3

Demand category (class)	Cumulative distribution
900	0.1428
1450	0.2857
2000	0.4285
2600	0.5715
3150	0.8571
3700	0.8571
4300	1.0000
4900	1.0000

Although the mean value of this distribution is 2507, the actual value of the mean order size is 1611. This is because of the correction factors associated with the distribution of proportion.(See table A2 - 9 for distributions of proportions).

TABLE A2 - 7

DISTRIBUTION OF PROPORTIONS

PRODUCT STYLE - 1

Shoe sizes	Distribution of proportions		
	Pattern-1	Pattern-2	Pattern-3
3	0.0334	0.0000	0.0000
3-1/2	0.0455	0.0362	0.0000
4	0.1072	0.2357	0.0000
4-1/2	0.1207	0.1560	0.0000
5	0.1690	0.1803	0.2622
5-1/2	0.1635	0.1639	0.1882
6	0.1560	0.1165	0.2026
6-1/2	0.0902	0.0227	0.0984
7	0.0681	0.0530	0.1514
7-1/2	0.0258	0.0094	0.0635
8	0.0193	0.0258	0.0325
Probability of occurrance	0.50	0.30	0.20
Correction factor for order size	1.00	0.41	0.32

TABLE A2 - 8

DISTRIBUTION OF PROPORTIONS

PRODUCT STYLE 2

Shoe sizes	Distribution of proportions		
	Pattern-1	Pattern-2	Pattern-3
2	0.0036	0.0000	0.0000
2-1/2	0.0040	0.0000	0.0000
3	0.0485	0.0658	0.1000
3-1/2	0.0465	0.0763	0.0683
4	0.1223	0.1295	0.1800
4-1/2	0.1151	0.1137	0.1850
5	0.1786	0.1547	0.1750
5-1/2	0.1479	0.1400	0.0450
6	0.1344	0.1399	0.2466
6-1/2	0.0847	0.0793	0.0000
7	0.0887	0.0611	0.0000
7-1/2	0.0130	0.0204	0.0000
8	0.0123	0.0187	0.0000
Probability of occurance	0.2857	0.6428	0.0714
Correction factor for order size	1.43	1.00	0.34

TABLE A2 - 9

DISTRIBUTION OF PROPORTIONS

PRODUCT STYLE 3

Shoe sizes	Distribution of proportions		
	Pattern-1	Pattern-2	Pattern-3
3	0.0619	0.0500	0.0000
3-1/2	0.0599	0.0000	0.0000
4	0.1199	0.1389	1.0000
4-1/2	0.1180	0.1667	0.0000
5	0.1422	0.1667	0.0000
5-1/2	0.1267	0.1667	0.0000
6	0.1425	0.1389	0.0000
6-1/2	0.0800	0.0833	0.0000
7	0.0770	0.0833	0.0000
7-1/2	0.0345	0.0000	0.0000
8	0.0367	0.0000	0.0000
Probability of occurrance	0.6363	0.2727	0.0909
Correction factor for order size	1.00	0.020	0.010

TABLE A2 - 10

LIST OF MOULDS

Product – style-1		Product – style-2		Product – style-3	
Nominal mould size	No. of moulds	Nominal mould size	No. of moulds	Nominal mould size	No. of moulds
2-1/2	1	2-1/2	1	2	1
3-1/2	1	3-1/2	1	3	1
4	2	4	1	4-1/2	1
5	2	5	1	5	2
5-1/2	2	5-1/2	2	6	1
6-1/2	2	6-1/2	1	7-1/2	1
7-1/2	1	7-1/2	1		

TABLE A2 - 11

DISTRIBUTION OF SETUP TIMES

Setup times (min)	Cumulative distribution
5	0.05
6	0.15
7	0.30
8	0.70
9	0.85
10	0.95
11	1.00

$\overline{X}$ = 8 min.

TABLE A 2- 12

Cost structure used in the study of strategies for capacity manipulation

Costs are given in a fictitious monetary unit called m.u.

1. Labour cost

Basic rate	- 3,942.00 m.u. for 40 hours/week
Next 4 hours/week (between 40 and 44 hours)	- premium of 25%
Next 4 hours/week (between 44 and 48 hours)	- premium of 50%
All after 48 hours/week	- premium of 100%

2. Cost of moulds

30,000.00 m.u. per mould with a depreciation period of 3 years.

This means a depreciation rate of 10,000.00 m.u. a year.

3. Cost of machines

2,500,000.00 m.u. per machine with a depreciation period of 5 years. This means a depreciation rate of 500,000.00 m.u. per year.

4. Inventory costs

Inventory is charged at 25% per year on the capital investment. The unit price of an item is equal to 10 m.u..This means that inventory is charged at a rate of 2.5 m.u. per year/per item.

APPENDIX 3

DETAILS OF THE DISTRIBUTIONS OF PROPORTIONS AND GOODNESS-OF-FIT TESTS ON THE EMPIRICAL DISTRIBUTIONS

A3 - 1 Details of the distribution of proportions.

The proportions of the total order which is required for each product size is called the distribution of proportions. The distribution used in the main core of the experiments was based on the distribution of proportions described in paragraph 2.2.3, and their numerical values are presented in the table below. It was assumed that all orders for all product styles would follow this distribution.

Product size	Proportions
2	0.003
2-1/2	0.004
3	0.043
3-1/2	0.055
4	0.122
4-1/2	0.133
5	0.169
5-1/2	0.139
6	0.149
6-1/2	0.078
7	0.076
7-1/2	0.026
8	0.021

A3 - 2 Goodness-of-fit tests on the empirical distributions

In order to decide on the distributions of interarrival times and order sizes a Kolmogoroff-Smirnof Goodness-of-fit test (1) was applied to the empirical data obtained in the industrial company. The analysis involves a much more comprehensive amount of data than the data used in the preliminary investigation. Four product styles are analysed where the first three correspond to the ones used in the preliminary investigation, and the fourth had been introduced after the first visit to the company. In the following tables the distributions of interarrival times and order sizes are tested against exponential distributions through the Kolmogoroff-Smirnof test.

(1) See SCHMIDT, J.W. and TAYLOR, R.E., 1970, Simulation and Analysis of Industrial Systems (USA, IRWIN INC.) p. 230

TABLE A3 - 1

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 1

$\overline{X}$ = 7.05 days

Interarrival times (days)	$S_T(x)$ observed cumulative distribution	$F_X(x)$ expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 1	0.0857	0.1322	0.0465
2	0.1999	0.2470	0.0471
3	0.3141	0.3466	0.0325
4	0.4569	0.4330	0.0239
5	0.5140	0.5080	0.0060
6	0.5997	0.5730	0.0267
7	0.6282	0.6295	0.0013
8	0.6567	0.6785	0.0218
9	0.6567	0.7083	0.0516
10	0.7424	0.7456	0.0032
11	0.7424	0.7781	0.0357
12	0.7709	0.8029	0.0320
13	0.7709	0.8277	0.0568 (+)
14	0.8280	0.8492	0.0212
15	0.8565	0.8681	0.0116
16	0.8850	0.8845	0.0005
17	0.9135	0.8988	0.0147
18	0.9135	0.9113	0.0022
19	0.9135	0.9222	0.0087
20	0.9135	0.9317	0.0182
21	0.9705	0.9412	0.0008
22	1.0000	0.9594	0.0074

(+) $D_{max} = 0.0568$ $D^*_{0.05} = 0.2266$ $D^*_{0.05} > D_{max}$

TABLE A3 - 2

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 2

$\bar{X}$ = 6.95 days

Interarrival times (days)	$S_T(x)$ observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 1	0.1428	0.1340	0.0088
2	0.2857	0.2501	0.0356
3	0.2857	0.3450	0.0593
4	0.3809	0.4311	0.0502
5	0.4761	0.5059	0.0298
6	0.5238	0.5709	0.0471
7	0.5238	0.6274	0.1036
8	0.5714	0.6764	0.1050 (+)
9	0.6666	0.7189	0.0523
10	0.7619	0.7559	0.0060
11	0.8571	0.7880	0.0691
12	0.8571	0.8159	0.0412
13	0.8571	0.8401	0.0170
14	0.8571	0.8611	0.0040
15	0.8571	0.8794	0.0223
16	0.8571	0.8953	0.0382
17	0.9047	0.9090	0.0043
18	0.9523	0.9210	0.0313
19	0.9523	0.9314	0.0209
20	0.9783	0.9404	0.0379
21	1.0000	0.9482	0.0518

(+) $D_{max} = 0.1050$ $D^*_{0.05} = 0.294$ $D^*_{0.05} > D_{max}$

TABLE A3 - 3

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 3

$\overline{X}$ = 6.37 days

Interarrival times (days)	$S_T(x)$ observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 1	0.1379	0.1452	0.0073
2	0.1379	0.2693	0.1314
3	0.2413	0.3793	0.1380
4	0.2758	0.4700	0.1942
5	0.2758	0.5475	0.2717 (+)
6	0.4482	0.6138	0.1656
7	0.5517	0.6704	0.1187
8	0.6206	0.7188	0.0982
9	0.7241	0.7602	0.0361
10	0.7241	0.7956	0.0715
11	0.7586	0.8258	0.0672
12	0.8965	0.8517	0.0448
13	0.8965	0.8738	0.0127
14	0.9310	0.8927	0.0383
15	1.0000	0.9088	0.0912

(+) $D_{max} = 0.2717$ $\quad D^*_{0.05} = 0.2413$ $\quad D_{max} > D^*_{0.05}$

Hypothesis of exponential distribution is rejected.

TABLE A3 - 4

DISTRIBUTION OF INTERARRIVAL TIMES

PRODUCT STYLE 4

$\bar{X}$ = 2.28 days

Interarrival times (days)	$S_T(x)$ Observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 1	0.3848	0.3551	0.0297
2	0.5899	0.5841	0.0058
3	0.7181	0.7317	0.0136
4	0.8078	0.8270	0.0192
5	0.8590	0.8884	0.0294 (+)
6	0.9359	0.9280	0.0079
7	0.9615	0.9536	0.0086
8	0.9743	0.9701	0.0042
9	0.9871	0.9807	0.0064
10	1.0000	0.9875	0.0125

(+) $D_{max} = 0.0294$ $D^*_{0.05} = 0.1539$ $D^*_{0.05} > D_{max}$

TABLE A3 - 5

DISTRIBUTION OF ORDER SIZES

PRODUCT STYLE 1

$\bar{X}$ = 1566

Demand category (class)	$S_T(x)$ Observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation $\lvert F_X(x) - S_T(x) \rvert$
0 - 600	0.4594	0.3183	0.1411
1200	0.7027	0.5353	0.1674 (+)
1800	0.7297	0.6832	0.0465
2400	0.7837	0.7840	0.0003
3000	0.7837	0.8528	0.0691
3600	0.8648	0.8996	0.0348
4200	0.8648	0.9316	0.0668
4800	0.8648	0.9534	0.0886
5400	1.0000	0.9682	0.0318

(+) $D_{max} = 0.1674$ $D^*_{0.05} = 0.2235$ $D^*_{0.05} > D_{max}$

TABLE A3 - 6

DISTRIBUTION OF ORDER SIZES

PRODUCT STYLE 2

$\overline{X} = 1410$

Demand category (class)	$S_T(x)$ Observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 600	0.4285	0.3127	0.1158
1200	0.4761	0.5276	0.0515
1800	0.5238	0.6753	0.1515 (+)
2400	0.7619	0.7768	0.0149
3000	0.9047	0.8466	0.0581
3600	0.9047	0.8946	0.0101
4200	0.9047	0.9275	0.0228
4800	0.9523	0.9502	0.0021
5400	1.0000	0.9657	0.0343

(+) $D_{max} = 0.1515$ $\quad D^*_{0.05} = 0.2940$ $\quad D^*_{0.05} > D_{max}$

TABLE A3 - 7

DISTRIBUTION OF ORDER SIZES

PRODUCT STYLE 3

$\overline{X}$ = 1897

Demand category (class)	$S_T(x)$ Observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0 - 600	0.2000	0.2717	0.0717
1200	0.3333	0.4688	0.1355 (+)
1800	0.5666	0.6128	0.0462
2400	0.6999	0.7178	0.0179
3000	0.8333	0.7943	0.0390
3600	0.8999	0.8501	0.0498
4200	0.8999	0.8907	0.0092
4800	0.8999	0.9204	0.0205
5400	1.0000	0.9420	0.0580

(+) D_{max} = 0.1355 $\quad D^*_{0.05}$ = 0.2400 $\quad D^*_{0.05} > D_{max}$

TABLE A3 - 8

DISTRIBUTION OF ORDER SIZES

PRODUCT STYLE 4

$\bar{X}$ = 1193

Demand category (class)	$S_T(x)$ Observed cumulative distribution	$F_X(x)$ Expected cumulative distribution	Deviation D $\lvert F_X(x) - S_T(x) \rvert$
0-600	0.3846	0.3934	0.0088
1200	0.6666	0.6321	0.0345 (+)
1800	0.7435	0.7768	0.0333
2400	0.8589	0.8646	0.0057
3000	0.9358	0.9179	0.0179
3600	0.9615	0.9502	0.0113
4200	0.9743	0.9698	0.0045
4800	0.9743	0.9816	0.0073
5400	1.0000	0.9888	0.0112

(+) D_{max} = 0.0345 $\quad D^*_{0.05}$ = 0.1539 $\quad D^*_{0.05} > D_{max}$

APPENDIX - 4

NUMERICAL RESULTS OF EXPERIMENTS WITH PRIORITY SCHEDULING RULES

A4 - 1 Introduction

As described in paragraph 6.1, the investigation with the priority scheduling rules consisted of thirty six experiments which were generated by testing each of six priority rules over six system configurations. Each one of the thirty six experiments was conducted in accordance with the sampling procedure described in paragraph 5.3.3, which means that six yields were generated for each one of the output variables.

Comparisons between the priority rules were made through the use of six measures of delivery performance, and five measures of internal behaviour. Four of the measures of delivery performance, viz. 'order late', 'tardiness of order', 'production late', and 'tardiness of production' were calculated for seven different values of the lead time D, meaning that 35 output variables were used, giving a total of 210 yields for every experiment. In tables 6.1 to 6.6 of chapter 6 the mean values and the result of the statistical tests for the six measures of delivery performance (for D = 8 days) were presented.

This appendix will present the mean values for all the five measures of internal behaviour (tables A4 - 1 to A4 - 6) and the mean values for the four measures of delivery performance which depend on the value of D, for D equal 8, 10, 12, 14, 16, 18 and 20. These values are the ones used in figures 6.1 to 6.24 of Chapter 6 and are presented here in tables A4 - 7 to A4 - 42.

In order to give an example of the procedure used in the statistical tests applied on the results of chapter 6, and to indicate the variability associated with the six yields obtained from the pair of antithetic runs, a selected example is presented in table A4 - 43.

TABLE A4 - 1

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration (I)

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	17.08	5.12	66.22	6.60	8892
FIFOMB	16.73	5.18	66.86	6.15	7882
SLACK	19.65	5.12	66.57	6.73	9319
SLACKM	17.91	5.15	66.46	6.10	8602
SPT	12.90	5.30	68.60	6.98	8406
SPTM	12.88	5.34	69.01	6.24	7217

TABLE A4 - 2

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration abc

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	87.81	4.65	79.27	5.23	13875
FIFOMB	85.71	4.58	78.19	4.58	13747
SLACK	90.18	4.65	79.40	5.26	14249
SLACKM	88.20	4.58	78.20	4.62	13603
SPT	67.91	4.87	83.60	5.86	18130
SPTM	63.33	4.71	81.15	4.65	17619

TABLE A4 - 3

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration def

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	14.29	5.71	68.38	6.30	9676
FIFOMB	13.22	5.44	65.03	5.05	9940
SLACK	16.40	5.72	68.50	6.29	9712
SLACKM	14.46	5.43	64.86	4.98	9746
SPT	8.92	5.89	70.77	6.51	10088
SPTM	8.88	5.70	68.15	5.21	10102

TABLE A4 - 4

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration ace

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	80.97	4.47	76.56	3.89	13604
FIFOMB	79.97	4.44	76.29	3.60	13689
SLACK	85.10	4.47	76.08	3.93	12936
SLACKM	83.47	4.44	75.71	3.60	12206
SPT	60.33	4.59	79.17	4.19	13877
SPTM	58.92	4.52	78.15	3.63	14920

TABLE A4 - 5

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration bdf

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	20.65	6.49	77.96	11.26	10727
FIFOMB	16.92	5.89	70.48	8.50	9805
SLACK	22.66	6.51	78.09	11.40	9712
SLACKM	17.70	5.86	70.31	8.42	10347
SPT	13.54	6.62	79.55	12.00	8862
SPTM	12.27	6.10	73.20	8.82	10882

TABLE A4 - 6

MEAN VALUE OF MEASURES OF INTERNAL BEHAVIOUR

System Configuration abcdef

PRIORITY RULE	NO. OF JOBS IN QUEUE	PROCESS CYCLE TIME (min.)	ACTUAL LOAD FACTOR (%)	IDLE TIME DUE TO SETUP (%)	REMAINING CONTENT
FIFOB	51.69	5.55	88.63	9.79	9451
FIFOMB	33.61	5.04	80.92	6.35	11397
SLACK	56.09	5.55	88.29	10.01	9100
SLACKM	38.35	5.03	80.73	6.24	10484
SPT	24.94	5.75	91.46	10.94	13616
SPTM	22.06	5.26	84.33	6.66	12038

TABLE A4 - 7

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	0.83	0.013	1.98	0.035
10	0.13	0.000	0.64	0.007
12	0.00	0.000	0.00	0.000
14	0.00	0.000	0.00	0.000
16	0.00	0.000	0.00	0.000
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 8

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	33.65	2.193	40.00	2.772
10	25.00	1.567	29.65	2.035
12	17.95	1.108	21.94	1.485
14	12.05	0.782	16.41	1.073
16	7.88	0.565	10.54	0.780
18	5.83	0.423	7.77	0.585
20	4.48	0.313	5.78	0.442

TABLE A4 - 9

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	5.90	0.148	12.01	0.318
10	2.11	0.053	4.63	0.122
12	0.89	0.022	2.05	0.053
14	0.25	0.005	0.81	0.015
16	0.06	0.002	0.20	0.003
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 10

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	30.89	1.887	36.52	2.432
10	22.56	1.312	27.54	1.753
12	15.19	0.903	20.05	1.250
14	9.23	0.630	12.83	0.890
16	6.28	0.463	8.74	0.657
18	5.06	0.345	7.03	0.493
20	3.91	0.247	5.34	0.358

TABLE A4 -11

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	11.22	0.323	18.22	0.562
10	5.19	0.137	9.19	0.253
12	2.05	0.053	3.78	0.107
14	0.89	0.018	1.89	0.043
16	0.25	0.005	0.59	0.013
18	0.06	0.000	0.20	0.002
20	0.00	0.000	0.00	0.000

TABLE A4 -12

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE FIFOB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	38.65	2.345	47.44	3.072
10	29.35	1.620	36.77	2.182
12	21.86	1.067	27.55	1.485
14	14.93	0.668	19.63	0.975
16	9.99	0.393	13.27	0.620
18	6.09	0.217	8.87	0.375
20	3.33	0.110	6.00	0.213

TABLE A4 - 13

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	0.96	0.022	1.81	0.042
10	0.32	0.005	0.57	0.010
12	0.06	0.000	0.04	0.000
14	0.00	0.000	0.00	0.000
16	0.00	0.000	0.00	0.000
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 14

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	31.98	2.025	38.83	2.598
10	23.71	1.430	28.36	1.883
12	16.53	0.993	21.25	1.362
14	10.64	0.698	14.53	0.975
16	6.92	0.512	9.11	0.713
18	5.38	0.385	7.46	0.543
20	4.29	0.285	5.61	0.403

TABLE A4 - 15

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	6.73	0.192	12.65	0.390
10	3.14	0.080	6.07	0.170
12	1.34	0.030	3.08	0.073
14	0.38	0.007	1.05	0.023
16	0.12	0.000	0.32	0.003
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 16

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	30.32	1.853	36.33	2.398
10	21.79	1.290	26.65	1.725
12	14.87	0.887	19.60	1.233
14	9.04	0.622	12.59	0.875
16	6.09	0.460	8.47	0.648
18	5.06	0.342	6.96	0.487
20	3.84	0.245	5.27	0.357

TABLE A4 - 17

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	8.01	0.262	14.24	0.490
10	4.48	0.122	8.23	0.238
12	1.92	0.050	3.60	0.098
14	0.76	0.018	1.83	0.043
16	0.32	0.005	0.76	0.008
18	0.06	0.000	0.02	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 18

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE FIFOMB

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	31.98	1.753	41.91	2.418
10	22.62	1.162	30.10	1.643
12	15.25	0.752	21.42	1.088
14	11.09	0.465	15.08	0.687
16	6.47	0.268	9.44	0.413
18	4.16	0.150	6.41	0.235
20	2.43	0.075	4.04	0.117

TABLE A4 - 19

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	0.77	0.013	1.12	0.022
10	0.13	0.000	0.38	0.003
12	0.00	0.000	0.00	0.000
14	0.00	0.000	0.00	0.000
16	0.00	0.000	0.00	0.000
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 20

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	37.95	2.552	42.12	2.930
10	28.84	1.838	32.21	2.145
12	21.21	1.312	23.53	1.560
14	14.48	0.923	16.61	1.127
16	9.42	0.667	11.30	0.828
18	7.17	0.495	8.67	0.625
20	5.19	0.365	6.49	0.465

TABLE A4 - 21

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	8.07	0.232	12.30	0.348
10	3.58	0.088	5.45	0.133
12	1.34	0.033	2.32	0.055
14	0.44	0.010	0.63	0.015
16	0.19	0.003	0.37	0.005
18	0.06	0.000	0.08	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 22

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	33.72	2.195	37.09	2.500
10	24.29	1.562	27.23	1.805
12	18.78	1.105	20.92	1.293
14	11.98	0.765	13.50	0.910
16	7.69	0.552	8.66	0.672
18	5.83	0.405	7.09	0.503
20	4.74	0.293	5.60	0.363

TABLE A4 - 23

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	13.91	0.438	19.47	0.633
10	6.73	0.205	9.79	0.307
12	3.46	0.085	5.10	0.133
14	1.15	0.028	1.78	0.045
16	0.51	0.010	0.90	0.018
18	0.12	0.002	0.29	0.003
20	0.00	0.000	0.00	0.000

TABLE A4 - 24

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE SLACK

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	47.56	3.222	54.23	3.712
10	36.79	2.328	41.75	2.695
12	28.20	1.630	31.97	1.898
14	20.83	1.103	24.41	1.292
16	15.19	0.715	17.54	0.850
18	10.44	0.437	12.11	0.533
20	6.66	0.248	7.89	0.308

TABLE A4 - 25

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	1.02	0.023	1.40	0.032
10	0.44	0.008	0.61	0.010
12	0.06	0.000	0.04	0.000
14	0.00	0.000	0.00	0.000
16	0.00	0.000	0.00	0.000
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 26

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	35.83	2.335	40.46	2.715
10	25.89	1.662	28.97	1.957
12	19.42	1.177	21.95	1.418
14	13.20	0.823	15.56	1.015
16	8.14	0.593	9.54	0.732
18	6.28	0.442	7.76	0.555
20	4.87	0.325	6.11	0.412

TABLE A4 - 27

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	8.14	0.272	12.89	0.430
10	4.87	0.125	7.53	0.202
12	2.05	0.045	3.31	0.080
14	0.64	0.013	1.35	0.027
16	0.12	0.003	0.17	0.005
18	0.06	0.000	0.08	0.002
20	0.00	0.000	0.00	0.000

TABLE A4 - 28

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	33.65	2.163	37.09	2.468
10	24.42	1.535	27.02	1.772
12	17.88	1.072	19.97	1.263
14	11.47	0.745	12.75	0.893
16	7.30	0.542	8.29	0.667
18	5.89	0.403	7.17	0.505
20	4.55	0.292	5.55	0.370

TABLE A4 - 29

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	8.91	0.308	14.01	0.513
10	5.06	0.150	8.31	0.265
12	2.69	0.065	4.50	0.117
14	0.76	0.020	1.66	0.043
16	0.38	0.008	0.84	0.013
18	0.12	0.002	0.14	0.002
20	0.00	0.000	0.00	0.000

TABLE A4 - 30

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE SLACKM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	36.54	2.175	44.65	2.662
10	27.82	1.483	33.77	1.828
12	19.23	0.975	24.31	1.208
14	14.03	0.613	17.50	0.747
16	8.52	0.358	10.44	0.435
18	5.45	0.203	6.56	0.247
20	2.94	0.112	3.58	0.133

TABLE A4 - 31

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	2.18	0.072	7.09	0.252
10	0.83	0.037	2.79	0.133
12	0.51	0.028	1.93	0.082
14	0.38	0.012	1.53	0.047
16	0.25	0.003	1.16	0.017
18	0.00	0.000	0.00	0.000
20	0.00	0.000	0.00	0.000

TABLE A4 - 32

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	21.21	1.392	37.24	3.713
10	15.12	1.013	28.95	3.013
12	11.28	0.743	23.21	2.465
14	8.46	0.545	19.05	2.018
16	6.28	0.402	15.07	1.660
18	4.74	0.302	12.50	1.370
20	3.65	0.220	9.96	1.120

TABLE A4 -33

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	9.48	0.527	22.72	1.477
10	6.79	0.357	16.80	1.060
12	4.04	0.240	10.71	0.765
14	2.69	0.170	7.52	0.573
16	1.92	0.122	5.41	0.433
18	1.34	0.090	4.15	0.332
20	1.02	0.067	3.22	0.255

TABLE A4 - 34

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	20.44	1.167	36.09	2.973
10	13.65	0.803	26.51	2.305
12	10.00	0.558	19.90	1.807
14	6.08	0.390	14.10	1.447
16	4.29	0.287	10.55	1.178
18	3.20	0.217	8.17	0.980
20	2.50	0.162	7.05	0.820

TABLE A4 - 35

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	15.64	1.182	34.10	3.275
10	12.56	0.898	27.58	2.628
12	9.80	0.673	21.93	2.108
14	7.24	0.498	16.99	1.685
16	5.76	0.380	13.42	1.370
18	4.74	0.282	11.28	1.112
20	3.72	0.203	9.21	0.898

TABLE A4 - 36

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE SPT

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	23.33	1.820	48.44	7.005
10	19.10	1.437	40.91	6.075
12	16.34	1.130	36.69	5.277
14	13.71	0.880	31.62	4.572
16	10.83	0.682	25.74	3.967
18	9.29	0.535	22.69	3.467
20	8.01	0.415	20.18	3.023

TABLE A4 - 37

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION (I) , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	2.24	0.082	5.82	0.228
10	1.08	0.043	2.97	0.128
12	0.89	0.020	2.66	0.072
14	0.32	0.008	1.17	0.030
16	0.12	0.002	0.52	0.010
18	0.06	0.000	0.21	0.002
20	0.00	0.000	0.00	0.000

TABLE A4 - 38

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abc , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	18.59	1.245	34.09	3.385
10	13.33	0.908	26.55	2.745
12	9.87	0.675	20.88	2.242
14	7.05	0.502	16.42	1.843
16	5.57	0.377	13.74	1.528
18	4.23	0.287	11.11	1.265
20	3.65	0.218	10.30	1.050

TABLE A4 - 39

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION def , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	8.46	0.385	19.04	0.952
10	5.25	0.233	12.48	0.610
12	3.07	0.143	7.62	0.390
14	1.92	0.088	5.15	0.248
16	1.34	0.050	3.89	0.150
18	0.57	0.027	1.85	0.087
20	0.38	0.017	1.22	0.052

TABLE A4 - 40

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION ace , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	19.03	1.127	34.81	2.947
10	13.33	0.785	26.80	2.297
12	9.74	0.543	20.43	1.792
14	6.66	0.375	15.44	1.412
16	4.29	0.262	10.64	1.125
18	3.20	0.190	8.38	0.925
20	2.56	0.138	7.31	0.763

TABLE A4 - 41

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION bdf , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	10.25	0.563	21.90	1.298
10	6.86	0.378	14.98	0.900
12	4.55	0.252	10.43	0.618
14	2.94	0.168	6.83	0.433
16	2.37	0.115	5.56	0.303
18	1.60	0.072	3.96	0.200
20	0.96	0.040	2.50	0.127

TABLE A4 - 42

MEAN VALUES OF MEASURES OF DELIVERY PERFORMANCE

SYSTEM CONFIGURATION abcdef , PRIORITY RULE SPTM

LEAD TIME D - (DAYS)	ORDERS LATE (%)	TARDINESS OF ORDERS	PRODUCTION LATE (%)	TARDINESS OF PRODUCTION
8	23.84	1.745	44.55	4.792
10	18.90	1.318	36.93	3.935
12	14.80	0.987	30.66	3.223
14	11.86	0.737	25.79	2.642
16	9.23	0.540	20.52	2.152
18	7.31	0.393	17.12	1.762
20	6.02	0.278	14.23	1.433

TABLE A4 - 43

Example of yields from individual antithetic pairs and statistical test on the results of priority scheduling rules study.

System configuration - abc

Measure of performance - 'Production delay'

Values of individual yields (antithetic pairs)					
FIFOB	FIFOMB	SLACK	SLACKM	SPTM	SPT
9.330	9.140	9.580	9.300	10.240	11.170
10.840	10.250	10.730	10.550	9.310	10.260
7.760	7.720	8.480	7.980	8.730	9.090
7.640	7.510	7.810	7.630	8.540	8.990
6.780	6.610	7.010	6.670	7.500	8.290
8.860	8.490	9.250	8.900	8.120	7.950
			MEAN $\bar{X}$		
8.535	8.287	8.810	8.505	8.740	9.292
			Differences from control		
0.248	0.000	0.523	0.218	0.453	1.005 **

$MS_e = 0.307$, $MS_{TR} = 0.720$

$F^*_{0.05} = 2.090$ $F^*_{0.01} = 2.600$

$t^*_{0.05} = 2.350$ $t^*_{0.01} = 3.110$

$$FS = \frac{MS_{TR}}{MS_e} = \frac{0.720}{0.307} = 2.34 \quad \therefore \quad FS > F^*_{0.01}$$

hypothesis of equal mean is rejected

$$D^*_{0.05} = t^*_{0.05} \sqrt{2MS_e/N} = 2.35 . \sqrt{2 * 0.307/6} = 0.75$$

$$D^*_{0.01} = t^*_{0.01} \sqrt{2MS_e/N} = 3.11 . \sqrt{2 * 0.307/6} = 0.99$$

(**) The only rule significantly different is SPT

APPENDIX 5

MEAN VALUES OF YIELDS FROM INDIVIDUAL EXPERIMENTS IN THE FACTORIAL DESIGN

A5 - 1 Introduction

As described in chapter 4, the factorial design consisted of 32 experiments. For each experiment 7 output variables were measured such that the main effects and interactions on these variables could be averaged over those 32 experiments. The results of the main effects and interactions have been already presented in chapter 7. In table A5 - 1, which follows, the 32 yields for each of the 7 output variables are presented.

FIGURE A5 - 1

MEAN VALUES OF YIELDS FROM INDIVIDUAL EXPERIMENTS IN THE FACTORIAL DESIGN

SYSTEM CONFIGURATION	AVERAGE NO. OF JOBS IN QUEUE	AVERAGE PROCESS CYCLE TIME	AVERAGE LOAD FACTOR	IDLE TIME DUE TO SETUP (%)	AVERAGE DELAY OF PRODUCTION	PERCENTAGE OF PRODUCTION LATE	TARDINESS INDEX OF PRODUCTION
(I)	16.563	5.193	68.212	6.217	2.998	1.927	0.045
af	50.043	5.322	90.155	7.887	6.783	30.693	1.652
bf	21.160	6.362	81.663	14.582	4.267	8.737	0.288
ab	58.843	5.452	93.567	10.835	5.198	15.788	0.492
cf	18.442	4.993	63.965	5.690	3.900	9.035	0.272
ac	72.945	4.462	77.703	3.695	6.438	25.512	1.470
bc	32.658	4.463	57.532	4.032	4.373	12.995	0.532
abcf	57.040	5.220	88.845	9.777	7.717	36.628	2.100
df	14.338	5.390	64.492	5.185	4.663	12.348	0.383
ad	34.773	4.888	79.392	4.743	5.447	17.370	0.552
bd	15.178	5.348	65.688	7.320	4.200	5.572	0.132
abdf	39.190	5.520	88.238	10.055	8.158	37.622	2.193
cd	25.043	4.178	51.162	1.832	5.310	16.693	0.648
acdf	34.915	4.767	77.380	4.282	7.512	34.942	1.823
bcdf	16.958	5.110	61.520	6.612	5.397	18.295	0.645
abcd	60.358	4.427	71.527	3.248	8.410	38.813	2.622
ef	16.385	5.727	73.232	7.903	3.755	7.203	0.187
ae	47.139	5.062	87.867	6.570	4.783	13.222	0.352
be	18.605	5.753	74.993	11.103	3.557	3.907	0.092
abef	54.102	5.750	96.668	13.135	7.897	37.913	2.248
ce	29.395	4.308	55.813	2.850	4.278	12.037	0.487
acef	48.995	4.918	83.460	5.978	6.832	30.027	1.583
bcef	19.860	5.497	70.493	10.122	4.372	10.708	0.370
abce	73.070	4.598	76.620	5.015	6.690	26.893	1.582
de	11.173	4.957	60.803	3.908	4.238	6.067	0.132
adef	30.500	5.195	83.503	5.585	7.202	30.667	1.642
bdef	14.403	5.997	71.963	9.678	5.415	15.970	0.522
abde	34.618	5.267	85.368	7.960	6.257	22.910	0.822
cdef	14.153	4.770	57.323	3.610	5.175	16.133	0.565
acde	52.900	4.328	69.712	2.283	8.210	36.998	2.448
bcde	22.582	4.282	52.458	2.483	5.527	16.962	0.708
abcdef	35.280	5.073	82.287	7.213	8.382	40.653	2.367