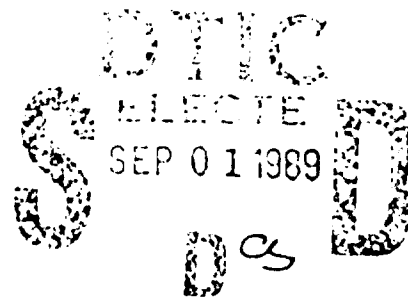


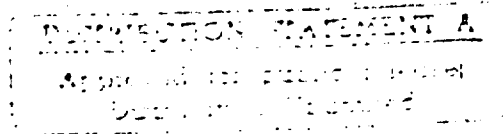
AD-A211 878



DEPARTMENT OF DEFENSE

**DEFENSE
LOGISTICS
AGENCY**

**MEDICAL ACQUISITION
SHELF—LIFE SYSTEM
(MASS)
DECISION SUPPORT MODEL
SYSTEMS DOCUMENTATION**



Cameron Station,
Alexandria, Virginia 22304-6100

MAY 1989

89 9 01 003

MEDICAL ACQUISITION SHELF-LIFE SYSTEM

(MASS)

DECISION SUPPORT MODEL SYSTEMS DOCUMENTATION

May 1989

Mr. Richard E. Baker, Mr. Rick L. Jernigan
Operations Research and Economic Analysis Office
Headquarters Defense Logistics Agency
Cameron Station, Alexandria, Virginia



DEFENSE LOGISTICS AGENCY

HEADQUARTERS
CAMERON STATION
ALEXANDRIA VIRGINIA 22304 6100

DLA-LO

FOREWORD

The Medical Acquisition Shelf-Life (MASS) Model is a decision aid to assist procurement analysts in evaluating alternative bids for stocked medical shelf-life items. MASS attempts to identify the best value bid by balancing longer shelf-life against higher purchase price in order to identify the bid with the lowest life cycle costs.

The objectives of this user's guide are to describe the model's features, instruct the user in using MASS, and explain the rationale of the model to vendors.

Christine L. Hall

ROGER C. ROY
Assistant Director
Office of Policy and Plans

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	



CONTENTS

<u>Title</u>	<u>Page</u>
Foreword.....	iii
Contents.....	v
List of Tables.....	vii
List of Figures.....	ix
I. Introduction.....	1
A. Background.....	1
B. Problem.....	1
C. Objectives.....	1
D. Scope.....	1
II. Conclusions.....	1
III. Recommendations.....	1
IV. Benefits.....	2
V. Implementation.....	2
VI. Model Features.....	2
VII. MASS Files.....	2
A. Program Files.....	3
B. Data Base Files.....	4
C. Index File.....	4
D. Configuration Files.....	5
E. Batch Files.....	5
F. Data Base Processing Files.....	5
VIII. Update Procedures.....	5
A. Update Data Annually.....	5
B. DRMS Data Call.....	5
C. Data Base Processing Programs.....	7
D. UPDATE.PRG.....	7
E. Updating DEFAULTS.DBF.....	9
F. Adding Or Deleting NSN Records.....	11
Appendices	
A. MASS.PRG Code.....	A-1
B. COMPUTE.PRG Code.....	B-1
C. NSNS.PRG Code.....	C-1
D. WRITE.PRG Code.....	D-1
E. SCREEN.PRG Code.....	E-1
F. RPT.PRG Code.....	F-1
G. UPDATE.PRG Code.....	G-1
H. M.BAT Code.....	H-1
I. CRASH.BAT Code.....	I-1
J. GOR.BAKER.JCL(P8016) Code.....	J-1
K. GOR.BAKER.JCL(P8016B) Code.....	K-1

LIST OF TABLES

<u>Table</u>	<u>Title</u>	<u>Page</u>
1	MASS Files.....	3
2	Telephone Numbers.....	7

LIST OF FIGURES

<u>Figure</u>	<u>Title</u>	<u>Page</u>
1	MASS Programs.....	4
2	Example of DRMS Data Call Letter.....	6
3	NSN Data Processing.....	8
4	Listing of DEFAULTS.DBF.....	9

I. INTRODUCTION

A. Background

In October 1987, LTG Russo hosted a conference with the Military Service Logistics Chiefs, the Service Surgeons General, and Office of Assistant Secretary of Defense (OASD) (Health Affairs), (Logistics), and (Environment) to discuss the Defense Logistics Agency (DLA) management of medical shelf-life materiel. Two of the action items resulting from this conference concerned the question of how much additional price the Department of Defense should pay to obtain a longer shelf-life period.

One of the action items was assigned to OASD (HA), the other to DLA. The OASD action item required that the MITRE model be evaluated to determine whether it could be used. The MITRE model was developed in the early eighties and was designed to address the additional price for longer shelf-life issue. The DLA action item required that an entirely new model be explored. Both OASD and DLA evaluated the MITRE model and concluded that the MITRE model had serious limitations and could not be used. As a result the Medical Acquisition Shelf-Life System (MASS) model was developed by the DLA Operations Research and Economic Analysis Management Support Office (DORO).

B. Problem. The Medical Acquisition Shelf-Life System (MASS) model is a decision aid to assist procurement analysts in evaluating alternative bids for stocked medical shelf-life items. The idea behind this model is not new. The model attempts to identify the best value bid by balancing longer shelf-life against higher purchase price. Life cycle costs are calculated for each bid by considering the purchase price, administrative buy, transportation, handling, storage, disposal, and replacement costs incurred for each bid.

C. Objectives. The objectives of this manual are to describe the data update procedures, the dBASE files, and the programs which produce the MASS files.

D. Scope. This model is limited to stocked medical shelf-life items, i.e. Replenishment Demand (Item Category Code 1 and P items).

II. CONCLUSIONS

A. MASS is a decision aid which can save money by considering the life cycle costs rather than just accepting the low bid.

B. MASS can save time by automating a difficult bid evaluation trade off decision.

C. Since MASS uses historical data to calculate costs, the data base must be updated periodically.

III. RECOMMENDATIONS

A. DORO recommends updating the MASS data annually.

B. Currently vendors indicate the shelf life of their products only if they do not meet the minimum shelf-life requirement. The Defense Personnel Support Center should modify their bid process in order to obtain shelf-life information for all medical shelf-life bids.

C. Procurement analysts should not be legally required to accept the MASS model recommendations. Analysts must be free to exercise their judgment.

IV. BENEFITS. The MASS model calculates cost avoidances by comparing the life cycle costs of each bid to the life cycle cost of the bid that normally would have been selected if the MASS model had not been used. Based on our initial testing of the model on actual evaluations, the average avoidance per evaluation was \$114,583. The Technical Services Branch, Technical Operations Division, Medical Directorate, Defense Personnel Support Center (DPSC-RSTH-21) estimated that the model would be used for 120 evaluations per year. If the recommendations of the MASS model are implemented, the Department of Defense would realize an estimated cost avoidance of up to \$13.75 million per year or a total net present value of \$137.5 million using the standard 10 percent discount factor.

V. IMPLEMENTATION. The MASS model has been implemented at the Defense Personnel Support Center to evaluate alternative bids for stocked medical shelf-life items. Prior to finalizing award of contracts, DPSC-RSTH-21 uses the MASS model to evaluate bids and make their recommendations to procurement. DPSC-RSTH-21 agreed to update the data according to the procedures specified in Section VIII.

VI. MODEL FEATURES

MASS allows the user to evaluate virtually an unlimited number of bids for an NSN at the same time. Additionally, it is easier to use because it provides NSN characteristics that are derived from historical data. In order to make running the model as easy as possible, the model provides default suggestions based on historical data for necessary input values. The model will display these default values and will give the user an opportunity to change them.

Also, the model will save the original bid information to facilitate investigating "what if" questions. For example, if the users want to investigate the effects on which bid would be accepted by changing the transportation and handling costs, the model will save the bid information for them so that they can rerun the model changing the transportation and handling costs as many times as they want. The user can then see what the affects of these changes are on the life cycle costs and ranking of the bids. Saving the users' bid information relieves them of reentering this information each time they want to change the defaults.

The model also gives the users the option to save their output to a disk file for use by other programs or packages such as dBASE, LOTUS, or ENABLE. If the MASS results are moved to these packages, reports can be customized, combined with other data, or graphed. This interface ability provides great flexibility. However, some familiarity with these packages is required.

VII. MASS FILES. MASS requires a collection of program, data base, index, configuration, and batch files to run properly. These files must be loaded on a hard disk to run MASS. If they are not loaded, see the Medical Acquisition Shelf-Life System (MASS) Decision Support Model User's Guide for loading instructions. Table 1 provides a list of all MASS files. Each file is explained in more detail in the following sections.

Table 1

MASS FILES

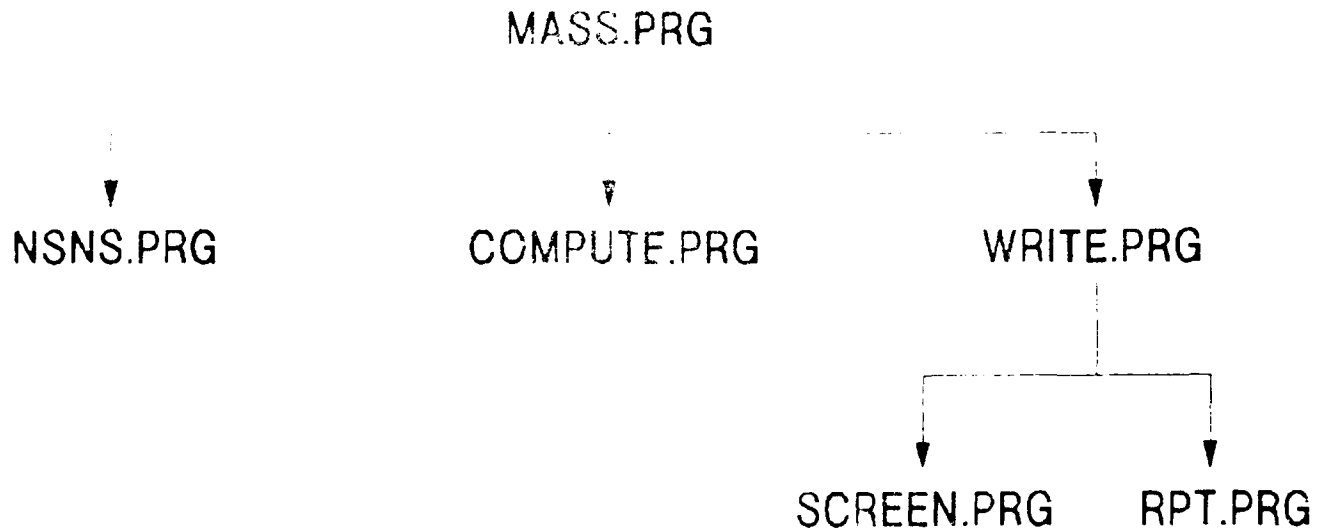
Program Files	MASS.PRG COMPUTE.PRG NSNS.PRG WRITE.PRG SCREEN.PRG RPT.PRG UPDATE.PRG
Data Base Files	NSN.DBF DEFAULTS.DBF RPT.DBF UCOST.DBF MASS123.DBF
Index File	NSN.NDX
Configuration Files	CONFIG.DB CONFIG.DB2
Batch Files	M.BAT CRASH.BAT
Data Base Processing Files	COR.BAFTER.ICL (P8016) COR.BAFTER.ICL (P8017)

module, prints the input data, and controls the writing of the Interface File. SCREEN.PRG displays the MASS results on the monitor screen. RPT.PRG prints the Medical Acquisition Shell Life System (MASS) Report.

UPDATE.PRG is a separate program which can run independently from the other MASS program modules. This program updates the National Stock Number (NSN) data base. It asks the users if they want to erase the current NSN data base. After receiving confirmation, this program erases the NSN dBASE file, then copies the new NSN text file into a dBASE format. Finally, this program indexes the NSN file to speed execution.

Figure 1

MASS PROGRAMS



Updating the MASS code is the responsibility of the Defense Logistics Agency, Operations Research and Economic Analysis Management Support Office (DORO). These program files should not be modified without the permission of the DORO office. Read only protection has been placed on these files. For your convenience, the dBASE code for these program files is listed in Appendices A through G.

B. Data Base Files. NSN.DBF, DEFAULTS.DBF, RPT.DBF, UCOST.DBF, and MASS123.DBF are the MASS data base files. NSN.DBF carries item characteristics and demand pattern data for each National Stock Number. DEFAULTS.DBF has only one record. It contains default values for NSNs not found in the NSN data, expandability data, system-wide costs, and

parameters for equations. RPT.DBF and UCOST.DBF are work files. They store bid information for the MASS screens and reports. MASS123.DBF is an interface file. Results can be stored in MASS123.DBF and be interfaced with other packages such as dBASE, LOTUS, or ENABLE.

C. Index File. NSN.NDX is the only indexed file in MASS. The NSN file was indexed to speed the execution of NSN searches. The NSN.NDX file corresponds to the NSN.DBF file. If records are added or deleted from the NSN.DBF file, then a new NSN.NDX file must be made by typing:

<INDEX ON NSN TO NSN>

D. Configuration Files. CONFIG.DB and CONFIG.DB2 are dBASE configuration files. They are automatically accessed when dBASE is executed. The configuration files establish the environment for dBASE.

E. Batch Files. M.BAT and CRASH.BAT are batch files. M.BAT removes read only protection from the program files, NSN.DBF and DEFAULTS.DBF, so that MASS can use them. M.BAT also renames the dBASE configuration files so that dBASE is properly configured for MASS. M.BAT then calls dBASE and executes the MASS programs. After MASS is finished, the M.BAT program returns the dBASE configuration to its normal setting, places read only protection back on the MASS files, and returns the computer to its original screen. If MASS is interrupted by a power failure, the M.BAT program will not properly change the configuration or replace the read only protection on the MASS files. CRASH.BAT restores the computer to its proper settings in the event of a power outage. The batch files are listed in Appendices H and I.

F. Data Base Processing Files. GOR.BAKER.JCL (P8016) and GOR.BAKER.JCL (P8016B) are mainframe program files which create the NSN data base. These files reside on the DORO01 disk pack of the DORAN computer. In the event that the disk pack files are accidentally erased, these files are backed up on tape number 007578. File # 1 on the tape is GOR.BAKER.JCL8016 and file # 2 is GOR.BAKER.JCL8016B. Copies of these programs are listed in Appendices J and K.

VIII. UPDATE PROCEDURES

A. Update Data Annually. Since MASS calculates life cycle costs based on historical data, this data must be periodically updated. We recommend that the data be updated annually.

B. DRMS Data Call. The first step in the data update process is to issue data calls to the Defense Reutilization Marketing Service (DRMS) and DORO. Figure 2 is an example of a DRMS data call letter. We suggest that you call DRMS and DORO before issuing the data calls. Table 2 lists telephone numbers for contacts as of October 1988.

SUBJECT: Request for Disposal Data

TO: Defense Reutilization Management Service
Federal Center
ATTN: DRMS-OD
Debbie Helton
North Washington Street
Battle Creek, Michigan 49017-3096

1. Reference: Telephone conversation between Ms. Debbie Helton, DRMS-OD, and Name, OFFICE-SYMBOL, Date, subject as above.
2. We request that you send us a 9 track 6,250 density EBCDIC IBM standard label tape. The tape should contain one year's worth of world-wide disposal receipt transactions (document identifier code 'XR1' and 'XR3' transactions).
3. Please mail the tapes to:

Address
ATTN: Name
City, State Zip Code
4. The point of contact for this request is Name, OFFICE-SYMBOL, Autovon phone number.

SIGNATURE BLOCK

Table 2

TELEPHONE NUMBERS

<u>For Help With:</u>	<u>Contact Person</u>	<u>Autovon Number</u>	<u>Office</u>
Shelf-Life Policies	Mike Pipan	284-6193	DLA-OSC
Extendibility/Expandability Statistics	LTC Horace Knight	343-7387	DMSB-11
Disposal Costs	Joann Stock	932-5902	DRMS-HP
Disposal Data Call Tapes	Debbie Helton	932-5966	DRMS-OD
MASS Systems Administration	Pat Humphrey	444-2129	DPSC-RSTH-21
MASS Programs and Data	Rick Baker	695-4918	DLA-DORO

C. Data Base Processing Programs. After you receive the tapes from DORO and DRMS, run the data processing program listed in Appendix K. This program may need to be adapted to run on your mainframe. This program matches the data on the two tapes and creates a data file called GOR.BAKER.P8016DB. This NSN data file should be downloaded to a diskette file called NSN.TXT.

D. UPDATE.PRG. This program copies the NSN.TXT file into a dBASE format. Figure 3 displays the files and programs for the NSN data processing. **WARNING**, UPDATE.PRG erases the NSN.DBF data prior to writing the new NSN data. Please follow these steps carefully to prevent data loss.

1. Enter dBASE
2. At the dot prompt, make a backup copy of NSN.DBF by typing:
`<COPY FILE NSN.DBF TO NSN2.DBF>`
3. When the dot prompt returns, execute UPDATE.PRG by typing:
`<DO UPDATE>`
4. Confirm that you want to zap NSN.DBF by typing: `<Y>`
5. Confirm that you want to overwrite NSN.NDX by typing: `<Y>`
6. Exit dBASE by typing: `<QUIT>`

E. Updating DEFAULTS.DBF

Since DEFAULTS.DBF has only one record, the easiest way to update this file is to edit it manually. Figure 4 lists the variables contained in DEFAULTS.DBF. The first ten variables listed in Figure 4 are missing value defaults. MASS uses these values when an NSN cannot be found on the NSN.DBF file. These default values are based on system average statistics which are printed by the GOR.BAKER.JCL (P8016) job. BADF, BPCM, BSHM, BSLM, and CONSTANT are updated using statistics printed by the GOR.BAKER.JCL (P8016B) job. Please save the output listing of these jobs until DEFAULTS.DBF is updated.

Figure 4

LISTING OF DEFAULTS.DBF

<u>Variable Name</u>	<u>Value</u>	<u>Description</u>
MSHM	31	Minimum Shelf Life
HAZ		Hazardous Code
UWEIGHT	16.04	Unit Weight
SLM	1.8	Safety Level Months
ADF	231	Annual Demand Frequency
PCM	7	Procurement Cycle Months
RPWR	0.0130	Rotatable Prepositioned War Reserve Probability
NPWR	0.0022	Nonrotatable Prepositioned War Reserve Probability
TRANH	0.1730	Transportation and Handling Cost
INAME	MISSING NSN VALUES	Item Name
PEXP	0.77	Probability of Expansion
EXPR	0.28	Expansion Rate
BUYCOST	250.00	Administrative Buy Cost
HDISP	8.3300	Hazardous Disposal Cost Per Pound
NDISP	8.1800	Nonhazardous Disposal Cost Per Pound
BADF	-0.00006151	Beta for Annual Demand Frequency
BPCM	0.00208200	Beta for Procurement Cycle Months
BSHM	-0.00114500	Beta for Shelf Life Months
BSLM	0.00131400	Beta for Safety Level Months
CONSTANT	0.09958500	Constant for Disposal Equation
HOLDING	0.11	Holding Cost per Inventory Dollar

PEXP and EXPR figures can be obtained from the Defense Medical Standardization Board (see Table 2 for telephone numbers). PEXP is the probability that the shelf-life can be expanded or extended by vendor. EXPR is the average proportion of increase in shelf-life if the stock is

expanded or extended. For example, if PEXP = .77 and EXPR = .28, this means that 77 percent of the items identified for extension are successfully extended. Of that 77 percent extended, their shelf-life, on the average, is increased by 28 percent.

Hazardous disposal cost per pound (HDISP) and nonhazardous disposal costs per pound (NDISP) are based on figures obtained from DRMS (see Table 2 for points of contact). The hazardous disposal administrative costs per pound (H) and nonhazardous administrative costs per pound (N) for Federal Supply Class 6505 are required. You will also need to know the average unit weight (W) and average unit price (P) for the medical data base. These figures can be obtained from the statistics printed by the GOR.BAKER.JCL (P8016) job. Once these are obtained, calculate the hazardous and nonhazardous costs with the following formulas:

$$\text{HDISP} = \text{H} + \text{P}/\text{W}$$

$$\text{NDISP} = \text{N} + \text{P}/\text{W}$$

The HAZ, INAME, BUYCOST, and HOLDING values in DEFAULTS.DBF are not likely to change. INAME is a default title used for the Item Name on the Medical Acquisition Shelf-Life System (MASS) Report when the NSN cannot be found on the NSN.DBF file. BUYCOST is the administrative buy cost specified by the Federal Acquisition Regulations. This figure does not change unless the law changes. HOLDING is the estimated holding cost per dollar of inventory. It includes 10 percent for the time value of money plus 1 percent for physical storage. Since HOLDING is a cost per dollar of inventory, this figure does not need to be updated for inflation. Data indicates that this value is relatively stable, but if you wish to change this figure, contact the Depot Operations Office at the Defense Logistics Agency Headquarters (DLA-OW) (see Table 2).

Once the new values are determined, the DEFAULTS.DBF can be easily edited by the following steps:

1. Change the directory to the dBASE directory by typing a back slash and the dBASE directory name for your computer. For most systems the dBASE directory is called "DBASE." If so, type: \DBASE.

2. Remove read only protection by typing:

<ATTRIB-R DEFAULTS.DBF>

3. Enter dBASE by typing: <DBASE>

4. At the dot prompt, type: <USE DEFAULTS>

5. Edit the file by typing: <EDIT>

6. The file can be changed by overwriting the highlighted values.

7. When finished, hit the <W> key while holding down the <CTRL> key to save the file.

8. Exit dBASE by typing: <QUIT>

F. Adding or Deleting NSN Records. NSN.DBF can be edited following procedures similiar to the 8 steps listed above, except in step 4 you would type: USE NSN. However, when you add or delete records from the NSN data base it is necessary to reindex the file for the MASS searches to work properly.

MASS.PRG Code

A-1

```

? '
? '
? 'BID ID          OFFERED   VALID   SHELF   INCLUDED'
                        UNIT PRICE (3-120)   BUY   LIFE?   IN OFFERED'
store bid to ubid
store price to uprice
store shm to ushm
store extend to uextend
store acost to uacost
@ 10, 0 get ubid
read
if ubid # '
    @ 10, 14 get uprice picture '#####.##'
    @ 10, col()+6 get ushm picture '@x ###' range 3,120
    @ 10, col()+2 get ubuyqty picture '@x #####' range 1,9999999
    @ 10, col()+9 get uextend picture 'L'
    @ 10, col()+2 get uacost picture '#####.##'
    if .not. unormal .and. .not. fnormal
        ?
        ? 'This is the bid that I normally would have taken (T or F)? '
        @ row(), col() get unormal picture 'L'
        read
        fnormal = unormal
        replace normal with unormal
        unormal = .f.
    else
        read
        replace normal with .f.
    endif
    replace bid with ubid
    replace price with uprice
    replace shm with ushm
    replace buyqty with ubuyqty
    replace extend with uextend
    replace acost with uacost
    do compute
    skip 1
endif
enddo
do while .not. eof()
    delete
    skip 1
enddo
do write
? 'I wish to run MASS again (T or F)? '
store .F. to continue
@row(), col() get continue picture 'L'
read
clear
enddo
clear
@ 12,23 say 'Thank you for using MASS.'
lines = 0
do while lines < 90
    lines = lines + 1
enddo
clear
quit

```

APPENDIX B
COMPUTE.PRG Code

```
use defaults
pdisp=badf*uadf+bpcm*upcm+bshm*ushm+bslm*uslm+constant
if pdisp < 0
  pdisp = 0
endif
if pdisp - unpwr > 1
  pdisp = 1 - unpwr
endif
if uextend
  eshm=ushm*pexp*expr+ushm
else
  eshm=ushm
endif
ushm=eshm-6.
if ushm < 3
  ushm = 3.0
endif
rshm=eshm-7.
if rshm < 3
  rshm = 3.0
endif
price=(uacost+uprice)*ubuyqty
thcost=Utranh*uuweight*ubuyqty
if uhaz = 'nl' .or. uhaz = ' '
  disp=ndisp*uuweight*ubuyqty+thcost+price*holding*eshm/12
  dispnr=ndisp*uuweight*ubuyqty+thcost
else
  disp=hdisp*uuweight*ubuyqty+thcost+price*holding*eshm/12
  dispnr=hdisp*uuweight*ubuyqty+thcost
endif
cum=0
if npwr > 0
  i=0
  do while i <= 120
    do case
      case i < 12
        disc=1.000
      case i < 24
        disc=.954
      case i < 36
        disc=.867
      case i < 48
        disc=.788
      case i < 60
        disc=.717
      case i < 72
        disc=.651
      case i < 84
        disc=.592
      case i < 96
        disc=.538
      case i < 108
        disc=.489
      case i < 120
        disc=.445
      otherwise
        disc=.405
    endcase
    cum=disc+cum
    i=i+ushm
  enddo
  term=(i-120)/ushm*.405
  cum=cum-term
endif
do case
  case eshm < 12
```

```

    edisc=1.000
    case eshm < 24
        edisc=.954
    case eshm < 48
        edisc=.788
    case eshm < 60
        edisc=.717
    case eshm < 72
        edisc=.651
    case eshm < 84
        edisc=.592
    case eshm < 96
        edisc=.538
    case eshm < 108
        edisc=.489
    case eshm < 120
        edisc=.445
    otherwise
        edisc=.405
endcase
do case
    case rshm < 12
        rdisc=1.000
    case rshm < 24
        rdisc=.954
    case rshm < 36
        rdisc=.867
    case rshm < 48
        rdisc=.788
    case rshm < 60
        rdisc=.717
    case rshm < 72
        rdisc=.651
    case rshm < 84
        rdisc=.592
    case rshm < 96
        rdisc=.538
    case rshm < 108
        rdisc=.489
    case rshm < 120
        rdisc=.445
    otherwise
        rdisc=.405
endcase
phold=(1-pdisp)*price*holding/2
nrhold=unpwr*price*holding*6.447
pcost=(1-Unpwr)*(price+thcost+buycost+phold+pdisp*dispc*edisc)
rpwrcoast=urpwr*thcost*rdisc
npwrcoast=unpwr*(price+thcost+buycost+dispnr)*cum+nrhold
lifecost=pcost+rpwrcoast+npwrcoast
use rpt
go recnum
replace ucost with lifecost/ubuyqty
replace lcost with lifecost
return

```

NSNS.PRG Code

```
store .T. to repeat
do while repeat
? 'Please enter the national stock number (NSN) for the bids'
? 'that you wish to evaluate '
? ' '
@ 7, 0 say 'nsn = ' GET UNSN PICTURE '9999999999999'
READ
use nsn index nsn
seek unsn
CLEAR
if eof()
? 'NSN not found. You may accept or modify the system '
? 'defaults for this NSN or enter a new NSN '
? 'System defaults are based on data base averages.'
USE DEFAULTS
store HAZ TO UHAZ
STORE INAME TO UINAME
UMSHM = mshm - 4
if umshm < 3
    umshm = 3
endif
STORE UWIGHT TO UUWEIGHT
store slm to uslm
store adf to uadf
store pcm to upcm
STORE RPWR TO URPWR
STORE NPWR TO UNPWR
STORE TRANH TO UTRANH
ELSE
? 'The data base information for this NSN may be accepted by'
? 'hitting enter or can be temporarily modified for this analysis.'
? 'Permanent data base updates may be made by browsing or '
? 'editing NSN.DBF.'
STORE HAZ TO UHAZ
STORE INAME TO UINAME
UMSHM = mshm - 4
if umshm < 3
    umshm = 3
endif
STORE UWIGHT TO UUWEIGHT
store slm to uslm
store adf to uadf
store pcm to upcm
STORE RPWR TO URPWR
STORE NPWR TO UNPWR
STORE TRANH TO UTRANH
ENDIF
wait
clear
? 'I wish to accept all these NSN characteristics (T or F)? '
store .T. to accept
@ row(), col() get accept picture 'L'
?
? 'NSN = '
?? unsn
?
? 'Hazardous or high disposal cost item code (blank means nonhaz.) = '
@ row(), col() say UHAZ picture '!!!'
?
? 'Item name = '
@ row + 1, col() say uiname
?
? 'Minimum shelf life required for this item (in days) = '
@ row + 1, col() say UMSHM picture '(60-999)'
?
```

```

@row(), col() say Uadfh picture '#####'
?
? 'Safety level months - '
@row(), col() say Uslm picture '###'
?
? 'Annual demand frequency - '
@row(), col() say Uadf picture '#####'
?
? 'Procurement cycle months (valid range 3 to 36) - '
@row(), col() say Upcm picture '@z ###'
?
? 'Probability of rotatable pwr - '
@row(), col() say urpwr picture '#.####'
?
? 'Probability of nonrotatable pwr - '
@row(), col() say unpwr picture '#.####'
?
? 'Transportation and handling $ per pound - '
@row(), col() say utranh picture '@z ###.####'
read
if accept
    return
endif
clear
? 'NSN - '
?? unsn
?
? 'Hazardous or high disposal cost item code (blank means nonhaz.) - '
@row(), col() get Uhaz picture '!!'
?
? 'Item name - '
@row(), col() get uiname
?
? 'Min shelf-life required for this NSN (valid range 3-120) - '
@row(), col() get UMSHM picture '###' range 3,120
?
? 'Unit weight - '
@row(), col() get Uuweight picture '@z #####' range .01,9999.99
?
? 'Safety level months - '
@row(), col() get Uslm picture '###'
?
? 'Annual demand frequency - '
@row(), col() get Uadf picture '#####'
?
? 'Procurement cycle months (valid range 3 to 36) - '
@row(), col() get Upcm picture '@z ###' range 3,36
?
? 'Probability of rotatable pwr - '
@row(), col() get urpwr picture '#.####'
?
? 'Probability of nonrotatable pwr - '
@row(), col() get unpwr picture '#.####'
?
? 'Transportation and handling $ per pound - '
@row(), col() get utranh picture '@z ###.####' range .0001,999.9999
?
? 'I wish to stop and enter a new NSN (T or F)? '
store .F. to repeat
@row(), col() get repeat picture 'L'
read
clear
enddo
return

```

APPENDIX D

WRITE.PRG Code

```

use rpt
go top
delete for bid = '
pack
sort to ucost on ucost
use ucost
go top
minprice=99999999.99
nlcost = 0
nbuyqty = 1
flag = .f.
do while .not. eof() .and. .not. flag
    if shm >= umshm .and. price < minprice
        minprice = price
        nlcost = lcost
        nbuyqty = buyqty
    endif
    if normal
        flag = .t.
        minprice = price
        nlcost = lcost
        nbuyqty = buyqty
    endif
    skip 1
enddo
go top
do while .not. eof()
    if nlcost = 0
        avoid = 0
    else
        avoid=buyqty*(nlcost/nbuyqty-lcost/buyqty)
    endif
    replace avoidance with avoid
    replace nsnout with unsn
    replace mshmout with umshm
    replace acceptbid with ' '
    replace haz with uhaz
    replace uweight with uuweight
    replace slm with uslm
    replace adf with uadf
    replace pcm with upcm
    replace rpwr with urpwr
    replace npwr with unpwr
    replace tranh with utranh
    skip 1
enddo
do screen
do rpt
?
set print off
set console on
clear
? 'I wish to print the data used in the analysis to compute'
? 'the life cycle costs?'
? 'Warning: This may print several pages of output.'
? 'I wish to print the input data (T or F)?'
store .T. to continue
@row(), col() get continue picture 'L'
read
if continue
    set print on
    set console off
    ? 'NSN Characteristics Used to Compute Life Cycle Costs'
    ?
    ? 'Date (mm/dd/yy) = '
    ?? date()

```

```

?? '    Time (hh:mm:ss) = '
?? time()
?
? 'Hazardous Storage Compatability Code = '
?? uhaz
? 'unit weight = '
?? uuweight
? 'safety level months = '
?? uslm
? 'annual demand frequency = '
?? uadf
? 'procurement cycle months = '
?? upcm
? 'rotatable PWR probability = '
?? urpwr
? 'nonrotatable PWR disposal probability = '
?? unpwr
? 'transportation and handling cost per pound = '
?? utranh
?
? 'Bid Extention and Additional Costs'
list off bid, extend, acost
use defaults
?
? 'System Default Values'
?
? 'Minimum Shelf-life Requirement = '
?? mshl
? 'Hazardous Storage Compatability Code = '
?? has
? 'Unit Weight = '
?? uweight
? 'Safety Level Months = '
?? slm
? 'Annual Demand Frequency = '
?? adf
? 'Procurement Cycle Months = '
?? pcm
? 'Rotatable PWR Probability = '
?? rpwr
? 'Nonrotatable PWR Disposal Probability = '
?? npwr
? 'Transportation and Handling Cost Per Pound = '
?? tranh
? 'Item Name = '
?? iname
? 'Probability of Shelf-life Extention or Expansion = '
?? pexp
? 'Percent Shelf-life Extention or Expansion = '
?? expr
? 'Administrative Cost of a Buy = '
?? buycost
? 'Hazardous Item Disposal Cost Per Pound = '
?? hdisp
? 'Nonhazardous Item Disposal Cost Per Pound = '
?? ndisp
? 'Beta Value for Annual Demand Frequency = '
?? badf
? 'Beta Value for Procurement Cycle Months = '
?? bpcm
? 'Beta Value for Shelf-life Months = '
?? bshl
? 'Beta Value for Safety Level Months = '
?? bslm
? 'Regression Constant = '
?? constant

```



```

? 'Annual holding cost per dollar of inventory = '
?? holding
?
endif
set console on
set print off
clear
? 'The results of this MASS analysis are stored in file MASS123.DBF.'
? 'This file has been formatted for LOTUS 123.'
? 'If you want to format this data for ENABLE, we suggest that you'
? 'use the translate option in LOTUS version 2 to write a .WKS file.'
? 'This .WKS file can then be read by ENABLE. We suggest this method'
? 'because some versions of ENABLE are not directly compatible with'
? 'dBASE III. See the Use manual for further details.'
?
? 'I want to erase the previous MASS123.DBF file (T or F)?'
store .F. to continue
@row(), col() get continue picture 'L'
read
if continue
store .N. to continue
? 'I am sure that I want to erase MASS123.DBF (T or F)?'
@row(), col() get continue picture 'L'
read
endif
if continue
use mass123
zap
use ucost
endif
store .T. to continue
? 'I want to append the current results to file MASS123.DBF (T or F)?'
@row(), col() get continue picture 'L'
read
if continue
use mass123
append from delimited from ucost
endif
clear
return

```

APPENDIX E

SCREEN.PRG Code

```

go top
set print off
set console on
do while .not. eof()
  clear
  ? '          Medical Acquisition Shelf-life System (MASS) Report'
  ?
  ? 'NSN - '
  ?? unsn
  ? 'Item Name - '
  ?? uiname
  ? 'Current Shelf-life Month Requirement - '
  ?? unshma
  ? 'Date (mm/dd/yy) - '
  ?? date()
  ?? '          Time (hh:mm:ss) - '
  ?? time()
  ?
  ? '          Ten Year          Te'
  ?? 'n Year          '
  ? '          Vendor Shelf Discounted          Disc'
  ?? 'ounted          Ten Year'
  ? '          Unit Life Unit Life          Buy          '
  ?? ' Life          Cost'
  ? 'Bid Id          Price Month Cycle Cost Quantity Cycle'
  ?? ' Cost          Avoidance'
  ?
  lines = 0
  do while lines < 11
    lines = lines + 1
    ? bid
    ?? ' '
    ?? price
    ?? ' '
    ?? shma
    ?? ' '
    ?? ucost
    ?? ' '
    ?? buyqty
    ?? ' '
    ?? lcost
    ?? ' '
    ?? avoidance
    skip 1
    if eof()
      lines = 52
    endif
  enddo
  wait
enddo
clear
return

```

APPENDIX F

RPT.PRG Code

```

clear
store .T. to continue
? 'I want to print this report (T or F)?'
@row(), col() get continue picture 'L'
read
if .not. continue
    return
endif
go top
set print on
set console off
do while .not. eof()
    ?
    ? '          Medical Acquisition Shelf-life System (MASS) Report'
    ?
    ? 'NSN - '
    ?? unsn
    ? 'Item Name - '
    ?? uiname
    ? 'Current Shelf-life Month Requirement - '
    ?? umshm
    ? 'Date (mm/dd/yy) - '
    ?? date()
    ?? '          Time (hh:mm:ss) - '
    ?? time()
    ?
    ? '
    ? '          Ten Year          Te'
    ?? 'n Year          ,          Te'
    ? '          Vendor Shelf Discounted Disc'
    ?? 'ounted          Ten Year'
    ? '          Unit Life Unit Life Buy '
    ?? ' Life          Cost'
    ? 'Bid Id          Price Month Cycle Cost Quantity Cycle'
    ?? ' Cost          Avoidance'
    ?
    lines = 0
    do while lines < 51
        lines = lines + 1
        ? bid
        ?? ' '
        ?? price
        ?? ' '
        ?? shm
        ?? ' '
        ?? ucost
        ?? ' '
        ?? buyqty
        ?? ' '
        ?? lcost
        ?? ' '
        ?? avoidance
        skip 1
        if eof()
            lines = 52
        endif
    enddo
    eject
enddo
?
return

```

APPENDIX G

UPDATE.PRG Code

USE NSN
ZAP
APPEND FROM NSN.TXT SDF
INDEX ON NSN TO NSN
USE

APPENDIX H

M.BAT Code

```
ECHO OFF
CD\DBASE
NOKEY
attrib -r nsn.dbf
attrib -r compute.prg
attrib -r mass.prg
attrib -r rpt.prg
attrib -r write.prg
attrib -r screen.prg
attrib -r defaults.dbf
rename config.db config2.db
rename config.db2 config.db
DBASE
rename config.db config.db2
rename config2.db config.db
attrib +r nsn.dbf
attrib +r compute.prg
attrib +r mass.prg
attrib +r rpt.prg
attrib +r write.prg
attrib +r screen.prg
attrib +r defaults.dbf
cd\
autoexec
```

APPENDIX I

CRASH.BAT Code

```
ECHO OFF
CD\DBASE
rename config.db config.db2
rename config2.db config.db
attrib +r nsn.dbf
attrib +r compute.prp
attrib +r mass.prp
attrib +r rpt.prp
attrib +r write.prp
attrib +r screen.prp
attrib +r defaults.dbf
cd\
autoexec
```

APPENDIX 1

GOR, BAKER, JCL(P8016) Code

```
//GORGOUID JOB (GO01,GO01), 'BAKER', MSGCLASS=V, CLASS=2, TIME=1440
//STEP1 EXEC PGM=BATCH204, REGION=3000K
//STEPLIB DD DSN=M204.R90.CRAMLIB, DISP=SHR
// DD DSN=M204.R90.LOAD, DISP=SHR
//CCASTAT DD DSN=M204.CCASTAT, DISP=SHR
//NSN DD DSN=GORD.NSN884, DISP=SHR
//DMRO1 DD DSN=GORD.DMRO881, DISP=SHR
//DMRO1A DD DSN=GORD.DMRO881A, DISP=SHR
//DMRO2 DD DSN=GORD.DMRO882, DISP=SHR
//DMRO2A DD DSN=GORD.DMRO882A, DISP=SHR
//DMRO3 DD DSN=GORD.DMRO883, DISP=SHR
//DMRO3A DD DSN=GORD.DMRO883A, DISP=SHR
//DMRO4 DD DSN=GORD.DMRO884, DISP=SHR
//DMRO4A DD DSN=GORD.DMRO884A, DISP=SHR
//CCAAUDIT DD SYSOUT=*
//CCAPRINT DD SYSOUT=*
//CCATEMP DD DSN=CCATEMP, DISP=(NEW,DELETE,DELETE), UNIT=WORKD,
// SPACE=(TRK,200)
//SYSPRINT DD SYSOUT=*
//OUTDATAN DD DSN=GOR.BAKER.P8016MRO, DISP=(NEW,PASS,DELETE),
// UNIT=TAPE, DCB=(RECFM=FB, LRECL=208, BLKSIZE=23296),
// LABEL=RETPD=180, VOL=(, , , 9)
//SYSOUT DD SYSOUT=*
//CCAIN DD *
NDIR=10, NFILES=10, LVTBL=2000, LNTBL=200, LQTBL=800, LRETBL=4000,
LSTBL=15528, MDKRD=5000000, MDKWR=5000000, MCNCT=250000, MUDD=2000000,
PAGESZ=6184, LFSCB=5000, UDDCCC=133, OUTMRL=133, MOUT=2000000, LFTBL=109432
LOGIN GOR6058
HUNDLEY
OPEN NSN
OPEN DMRO1
OPEN DMRO2
OPEN DMRO3
OPEN DMRO4
BEGIN
IMAGE NSNOUT
OUT.NSN IS STRING LEN 13
OUT.HAZ IS STRING LEN 2
OUT.INAME IS STRING LEN 19
OUT.LDISP IS STRING LEN 5 PAD '0' JUSTIFY RIGHT
OUT.SHLM IS STRING LEN 2 PAD '0' JUSTIFY RIGHT
OUT.SPRQC IS STRING LEN 1
OUT.UCUBE IS STRING LEN 6 PAD '0' JUSTIFY RIGHT
OUT.UWEIGHT IS STRING LEN 6 PAD '0' JUSTIFY RIGHT
OUT.UPRICE IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.SLM IS STRING LEN 3 PAD '0' JUSTIFY RIGHT
OUT.BOQ IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.IAQ IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.MADQ IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.OWRMRP IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.ADQ IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.ADF IS STRING LEN 7 PAD '0' JUSTIFY RIGHT
OUT.PCM IS STRING LEN 3 PAD '0' JUSTIFY RIGHT
OUT.ROP IS STRING LEN 7 PAD '0' JUSTIFY RIGHT
OUT.QFD IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.QFR IS STRING LEN 9 PAD '0' JUSTIFY RIGHT
OUT.RDN IS STRING LEN 15
OUT.NSHIP IS STRING LEN 3 PAD '0' JUSTIFY RIGHT
OUT.DIC IS STRING LEN 3
OUT.PROJ IS STRING LEN 3
OUT.COND IS STRING LEN 1
```

OUT.SERV IS STRING LEN 1	00680002
OUT.IPG IS STRING LEN 1 PAD '0' JUSTIFY RIGHT	00690002
OUT.PPZONE IS STRING LEN 1	00700002
OUT.DEST IS STRING LEN 2	00710002
OUT.POE IS STRING LEN 3	00720002
OUT.MODE IS STRING LEN 1	00730002
OUT.TRANCOST IS STRING LEN 7 PAD '0' JUSTIFY RIGHT	00740002
OUT.SHIPDATE IS STRING LEN 5 PAD '0' JUSTIFY RIGHT	00750002
OUT.QTYOFF IS STRING LEN 5 PAD '0' JUSTIFY RIGHT	00760002
OUT.MROCUBE IS STRING LEN 5 JUSTIFY RIGHT PAD '0'	00770002
OUT.MROWGT IS STRING LEN 6 JUSTIFY RIGHT PAD '0'	00780002
END IMAGE	00790002
OPEN DATASET OUTDATAN FOR OUTPUT	00800002
PREPARE IMAGE NSNOUT	00810002
FIND2: IN NSN	00820002
FD SHLM IS GT 0	00830002
DSC = M	00840002
ICC = 1 OR P	00850002
END FIND	00860002
FR1: FR FIND2	00870002
%NSNOUT:OUT.NSN = NSN	00880002
%NSNOUT:OUT.HAZ = HZSC	00890002
%NSNOUT:OUT.INAME = ITEMNAME	00891002
%NSNOUT:OUT.LDISP = LDDATE	00892002
%NSNOUT:OUT.SHLM = SHLM	00893002
%NSNOUT:OUT.SPRQC = SPRQC	00894002
%NSNOUT:OUT.UCUBE = UCUBE * 1000	00895002
%NSNOUT:OUT.UWEIGHT = UWEIGHT * 100	00896002
%NSNOUT:OUT.UPRICE = UPRICE * 100	00897002
%NSNOUT:OUT.SLM = SLM * 10	00898002
%NSNOUT:OUT.BOQ = BOQ	00899002
%NSNOUT:OUT.IAQ = IAQ	00899102
%NSNOUT:OUT.MADQ = MADQ * 10	00899202
%NSNOUT:OUT.OWRMRP = OWRMRP	00899302
%NSNOUT:OUT.ADQ = ADQ	00899402
%NSNOUT:OUT.ADF = ADF	00899502
%NSNOUT:OUT.PCM = PCM	00899602
%NSNOUT:OUT.ROP = ROP	00899702
%NSNOUT:OUT.QFD = QFD	00899802
%NSNOUT:OUT.QFR = QFR	00899902
FIND3: IN DMR01, DMR02, DMR03, DMR04 FD NSNORPART = %NSNOUT:OUT.NSN	00900002
END FIND	00900102
FR2: FR FIND3	00900202
IF TOTCUBE > 0 THEN	00900302
%MROCUBE = \$ROUND(TOTCUBE)	00900402
ELSE	00900502
%MROCUBE = \$ROUND(CUBE * QTY OFF)	00900602
END IF	00900702
IF TOTWGT > 0 THEN	00900802
%MROWGT = TOTWGT	00900902
ELSE	00901002
%MROWGT = \$ROUND(WGTXQTY,0)	00901102
END IF	00901202
IF %MROWGT = 0 THEN	00901302
%MROWGT = 1	00901402
END IF	00901502
%NSNOUT:OUT.RDN = RDN	00901602
%NSNOUT:OUT.NSHIP = NUM SHIPS	00901702
%NSNOUT:OUT.DIC = DIC	00901802
%NSNOUT:OUT.PROJ = PROJ CODE	00901902
%NSNOUT:OUT.COND = COND CODE	00902002
%NSNOUT:OUT.SERV = SERV IND	00902102
%NSNOUT:OUT.IPG = IPG	00902202
%NSNOUT:OUT.PPZONE = PPZONE	00902302
%NSNOUT:OUT.DEST = DEST CODE	00902402
	00902502

%NSNOUT:OUT.POE = POE	00902602
%NSNOUT:OUT.MODE = MODE	00902702
%NSNOUT:OUT.TRANCOST = TRANS COST * 100	00902802
%NSNOUT:OUT.SHIPDATE = DATE SHIP	00902902
%NSNOUT:OUT.QTYOFF = QTY OFF	00903002
%NSNOUT:OUT.MROCUBE = %MROCUBE	00903102
%NSNOUT:OUT.MROWGT = %MROWGT	00903202
WRITE IMAGE NSNOUT ON OUTDATAN	00903302
END FOR	00903402
END FOR	00903502
END	00903602
CLOSE ALL	00903702
EOJ	00903802
/*	00903902
//STEP2 EXEC PGM=IERRC000,REGION=990K,PARM='CORE=MAX'	00904002
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR	00904102
//SYSUDUMP DD SYSOUT=*	00904202
//SORTMSG DD SYSOUT=*	00904302
//SYSOUT DD SYSOUT=*	00904402
//SORTIN DD DSN=GOR.BAKER.P8016MRO,	00904502
// DISP=(SHR)	00904602
//SORTOUT DD DSN=GOR.BAKER.P8016MS,	00904702
// UNIT=TAPE,DISP=(NEW,CATLG,DELETE),	00904802
// LABEL=RETPD=20C	00904902
//SORTWK01 DD UNIT=WORKD,SPACE=(TRK,913)	00905002
//SORTWK02 DD UNIT=WORKD,SPACE=(TRK,913)	00905102
//SORTWK03 DD UNIT=WORKD,SPACE=(TRK,913)	00905202
//SORTWK04 DD UNIT=WORKD,SPACE=(TRK,913)	00905302
//SORTWK05 DD UNIT=WORKD,SPACE=(TRK,913)	00905402
//SORTWK06 DD UNIT=WORKD,SPACE=(TRK,913)	00905502
//SYSIN DD *	00905602
SORT FIELDS=(1,13,CH,A)	00905702
END	00905802
/*	00905902
//STEP3 EXEC SPSSX,CYLS='2,200'	00906002
TITLE 'GOR.BAKER.P8016MRO FREQUENCY REPORT'	00906102
DATA LIST FILE = INFILE /	00906202
LDDATE 35-39	00906302
SHM 40-41	00906402
SPRQC 42 (A)	00906502
UCUBE 43-48 (3)	00906602
UWEIGHT 49-54 (2)	00906702
UPRICE 55-63 (2)	00906802
SLM 64-66 (1)	00906902
BOQ 67-75	00907002
IAQ 76-84	00907102
MADQ 85-93 (1)	00907202
OWRMRP 94-102	00907302
ADQ 103-111	00907402
ADF 112-118	00907502
PCM 119-121	00907602
ROP 122-128	00907702
QFD 129-137	00907802
QFR 138-146	00907902
RDN1 147 (A)	00908002
NSHIP 162-164	00908102
DIC 165-167 (A)	00908202
PROJ 168-170 (A)	00908302
PROJ1 168 (A)	00908402
COND 171 (A)	00908502
IPG 173 (A)	00908602
PPZONE 174 (A)	00908702
DEST 175-176 (A)	00908802
POE 177-179 (A)	00908902
MODE 180 (A)	00909002
TRANCOST 181-187 (2)	00909102

SHIPYR 188-189	00909202
QTYOFF 193-197	00909302
MROCUBE 198-202	00909402
MROWGT 203-208	00909502
MISSING VALUES LDDATE SHM UCUBE TO QFR TRANCOST TO MROWGT (0)	00909602
RECODE RDN1 ('A', 'B', 'C', 'W' = 'A')	00909702
('N', 'P', 'Q', 'R', 'V', 'L', 'M', 'Z' = 'N')	00909802
('D', 'E', 'F', 'J' = 'F')	00909902
STRING PWR (A8)	00910002
DO IF RDN1 EQ 'A' AND (PROJ = '3AA' OR PROJ = 'WR2')	00910102
COMPUTE PWR = 'A 97%	00910202
ELSE IF RDN1 EQ 'N' AND (PROJ1 = 'P')	00910302
COMPUTE PWR = 'N 95%	00910402
ELSE IF RDN1 EQ 'N' AND (PROJ = '714')	00910502
COMPUTE PWR = 'N 0%	00910602
ELSE IF RDN1 EQ 'F' AND (PROJ = '262' OR PROJ = '596')	00910702
COMPUTE PWR = 'F 0%	00910802
ELSE IF RDN1 EQ 'F' AND (PROJ NE ' ')	00910902
COMPUTE PWR = 'F 75%	00911002
ELSE	00911102
COMPUTE PWR = ' NONE '	00911202
END IF	00911302
RECODE MPOWGT (0 THRU 69.99=69)	00911402
(70 THRU 499.99=499)	00911502
(500 THRU 999.99=999)	00911602
(1000 THRU 1999.99=1999)	00911702
(2000 THRU 4999.99=4999)	00911802
(5000 THRU 10000=10000)	00911902
(10000 THRU HI=10001) INTO WGTCAT	00912002
STRING REFRIG (A8)	00912102
RECODE SPRQC ('U', '3', '4', '5', '6' = ' REFRIG ')	00912202
('0', ' ', 'Z' = 'NOREFRIG')	00912302
(ELSE = ' NO REQ ') INTO REFRIG	00912402
BREAKDOWN QTYOFF BY PWR	00912502
OPTIONS 6	00912602
CONDESCRIPTIVE TRANCOST TO MROWGT	00912702
FREQUENCIES VARIABLES= RDN1 TO DIC COND TO MODE SHIPYR	00912802
FINISH	00912902
//INFILE DD DSN=GOR.BAKER.P8016MRO,	00913002
// DISP=SHR	00913102
//*	00913202
//STEP4 EXEC PGM=ATJ91U	00913302
//SYSPRINT DD SYSOUT=*	01980001
//SYSUT1 DD DSN=GOR.BAKER.P8016MS,DISP=SHR	01990001
//SYSUT2 DD DSN=GOR.BAKER.P8016AD,DISP=(NEW,CATLG,DELETE),	02000001
UNIT=3380,VOL=SER=WORKWS,SPACE=(TRK,(130,160),RLSE),	02010001
DCB=(LRECL=36,BLKSIZE=24444,RECFM=FB)	02020001
DD *	02030001
* DEFAULT PWR MRO QTY PROPORTION = 0 IN OUTPUT COLUMNS 14-18	02040001
* DEFAULT PWR MRO DISPOSAL QTY PROPORTION = 0 IN OUTPUT COLUMNS 19-23	02050001
* DEFAULT TRANS COST PER POUND = \$0.082 IN OUTPUT COLUMNS 24-29	02060001
* DEFAULT HANDLING COST PER POUND = \$0.091 IN OUTPUT COLUMNS 30-36	02070001
* DEFAULT MODE G TRANS COST PER POUND = \$0.24	02080001
* DEFAULT MODE H TRANS COST PER POUND = \$1.16	02090001
* MAINLINE	02100001
IF EQJ	02110001
GO 2 EOJ ENDIF	02120001
IF SYSUT1 = 1	02130001
M 1 TO H1,13 ENDIF	02140001
M 1 TO H1,13	02150001
M 1 TO H1,13	02160001
M 1 TO H1,13	02170001
M 1 TO H1,13	02180001
M 1 TO H1,13	02190001
M 1 TO H1,13	02200001
M 1 TO H1,13	02210001
M 1 TO H1,13	02220001
M 1 TO H1,13	02230001
M 1 TO H1,13	02240001
M 1 TO H1,13	02250001
M 1 TO H1,13	02260001
M 1 TO H1,13	02270001
M 1 TO H1,13	02280001
M 1 TO H1,13	02290001
M 1 TO H1,13	02300001
M 1 TO H1,13	02310001
M 1 TO H1,13	02320001
M 1 TO H1,13	02330001
M 1 TO H1,13	02340001
M 1 TO H1,13	02350001
M 1 TO H1,13	02360001
M 1 TO H1,13	02370001
M 1 TO H1,13	02380001
M 1 TO H1,13	02390001
M 1 TO H1,13	02400001
M 1 TO H1,13	02410001
M 1 TO H1,13	02420001
M 1 TO H1,13	02430001
M 1 TO H1,13	02440001
M 1 TO H1,13	02450001
M 1 TO H1,13	02460001
M 1 TO H1,13	02470001
M 1 TO H1,13	02480001
M 1 TO H1,13	02490001
M 1 TO H1,13	02500001
M 1 TO H1,13	02510001
M 1 TO H1,13	02520001
M 1 TO H1,13	02530001
M 1 TO H1,13	02540001
M 1 TO H1,13	02550001
M 1 TO H1,13	02560001
M 1 TO H1,13	02570001
M 1 TO H1,13	02580001
M 1 TO H1,13	02590001
M 1 TO H1,13	02600001
M 1 TO H1,13	02610001
M 1 TO H1,13	02620001
M 1 TO H1,13	02630001
M 1 TO H1,13	02640001
M 1 TO H1,13	02650001
M 1 TO H1,13	02660001
M 1 TO H1,13	02670001
M 1 TO H1,13	02680001
M 1 TO H1,13	02690001
M 1 TO H1,13	02700001
M 1 TO H1,13	02710001
M 1 TO H1,13	02720001
M 1 TO H1,13	02730001
M 1 TO H1,13	02740001
M 1 TO H1,13	02750001
M 1 TO H1,13	02760001
M 1 TO H1,13	02770001
M 1 TO H1,13	02780001
M 1 TO H1,13	02790001
M 1 TO H1,13	02800001
M 1 TO H1,13	02810001
M 1 TO H1,13	02820001
M 1 TO H1,13	02830001
M 1 TO H1,13	02840001
M 1 TO H1,13	02850001
M 1 TO H1,13	02860001
M 1 TO H1,13	02870001
M 1 TO H1,13	02880001
M 1 TO H1,13	02890001
M 1 TO H1,13	02900001
M 1 TO H1,13	02910001
M 1 TO H1,13	02920001
M 1 TO H1,13	02930001
M 1 TO H1,13	02940001
M 1 TO H1,13	02950001
M 1 TO H1,13	02960001
M 1 TO H1,13	02970001
M 1 TO H1,13	02980001
M 1 TO H1,13	02990001
M 1 TO H1,13	03000001
M 1 TO H1,13	03010001
M 1 TO H1,13	03020001
M 1 TO H1,13	03030001
M 1 TO H1,13	03040001
M 1 TO H1,13	03050001
M 1 TO H1,13	03060001
M 1 TO H1,13	03070001
M 1 TO H1,13	03080001
M 1 TO H1,13	03090001
M 1 TO H1,13	03100001
M 1 TO H1,13	03110001
M 1 TO H1,13	03120001
M 1 TO H1,13	03130001
M 1 TO H1,13	03140001
M 1 TO H1,13	03150001
M 1 TO H1,13	03160001
M 1 TO H1,13	03170001
M 1 TO H1,13	03180001
M 1 TO H1,13	03190001
M 1 TO H1,13	03200001
M 1 TO H1,13	03210001
M 1 TO H1,13	03220001
M 1 TO H1,13	03230001
M 1 TO H1,13	03240001
M 1 TO H1,13	03250001
M 1 TO H1,13	03260001
M 1 TO H1,13	03270001
M 1 TO H1,13	03280001
M 1 TO H1,13	03290001
M 1 TO H1,13	03300001
M 1 TO H1,13	03310001
M 1 TO H1,13	03320001
M 1 TO H1,13	03330001
M 1 TO H1,13	03340001
M 1 TO H1,13	03350001
M 1 TO H1,13	03360001
M 1 TO H1,13	03370001
M 1 TO H1,13	03380001
M 1 TO H1,13	03390001
M 1 TO H1,13	03400001
M 1 TO H1,13	03410001
M 1 TO H1,13	03420001
M 1 TO H1,13	03430001
M 1 TO H1,13	03440001
M 1 TO H1,13	03450001
M 1 TO H1,13	03460001
M 1 TO H1,13	03470001
M 1 TO H1,13	03480001
M 1 TO H1,13	03490001
M 1 TO H1,13	03500001
M 1 TO H1,13	03510001
M 1 TO H1,13	03520001
M 1 TO H1,13	03530001
M 1 TO H1,13	03540001
M 1 TO H1,13	03550001
M 1 TO H1,13	03560001
M 1 TO H1,13	03570001
M 1 TO H1,13	03580001
M 1 TO H1,13	03590001
M 1 TO H1,13	03600001
M 1 TO H1,13	03610001
M 1 TO H1,13	03620001
M 1 TO H1,13	03630001
M 1 TO H1,13	03640001
M 1 TO H1,13	03650001
M 1 TO H1,13	03660001
M 1 TO H1,13	03670001
M 1 TO H1,13	03680001
M 1 TO H1,13	03690001
M 1 TO H1,13	03700001
M 1 TO H1,13	03710001
M 1 TO H1,13	03720001
M 1 TO H1,13	03730001
M 1 TO H1,13	03740001
M 1 TO H1,13	03750001
M 1 TO H1,13	03760001
M 1 TO H1,13	03770001
M 1 TO H1,13	03780001
M 1 TO H1,13	03790001
M 1 TO H1,13	03800001
M 1 TO H1,13	03810001
M 1 TO H1,13	03820001
M 1 TO H1,13	03830001
M 1 TO H1,13	03840001
M 1 TO H1,13	03850001
M 1 TO H1,13	03860001
M 1 TO H1,13	03870001
M 1 TO H1,13	03880001
M 1 TO H1,13	03890001
M 1 TO H1,13	03900001
M 1 TO H1,13	03910001
M 1 TO H1,13	03920001
M 1 TO H1,13	03930001
M 1 TO H1,13	03940001
M 1 TO H1,13	03950001
M 1 TO H1,13	03960001
M 1 TO H1,13	03970001
M 1 TO H1,13	03980001
M 1 TO H1,13	03990001
M 1 TO H1,13	04000001
M 1 TO H1,13	04010001
M 1 TO H1,13	04020001
M 1 TO H1,13	04030001
M 1 TO H1,13	04040001
M 1 TO H1,13	04050001
M 1 TO H1,13	04060001
M 1 TO H1,13	04070001
M 1 TO H1,13	04080001
M 1 TO H1,13	04090001
M 1 TO H1,13	04100001
M 1 TO H1,13	04110001
M 1 TO H1,13	04120001
M 1 TO H1,13	04130001
M 1 TO H1,13	04140001
M 1 TO H1,13	04150001
M 1 TO H1,13	04160001
M 1 TO H1,13	04170001
M 1 TO H1,13	04180001
M 1 TO H1,13	04190001
M 1 TO H1,13	04200001
M 1 TO H1,13	04210001
M 1 TO H1,13	04220001
M 1 TO H1,13	04230001
M 1 TO H1,13	04240001
M 1 TO H1,13	04250001
M 1 TO H1,13	04260001
M 1 TO H1,13	04270001
M 1 TO H1,13	04280001
M 1 TO H1,13	04290001
M 1 TO H1,13	04300001
M 1 TO H1,13	04310001
M 1 TO H1,13	04320001
M 1 TO H1,13	04330001
M 1 TO H1,13	04340001
M 1 TO H1,13	04350001
M 1 TO H1,13	04360001
M 1 TO H1,13	04370001
M 1 TO H1,13	04380001
M 1 TO H1,13	04390001
M 1 TO H1,13	04400001
M 1 TO H1,13	04410001
M 1 TO H1,13	04420001
M 1 TO H1,13	04430001
M 1 TO H1,13	04440001
M 1 TO H1,13	04450001
M 1 TO H1,13	04460001
M 1 TO H1,13	04470001
M 1 TO H1,13	04480001
M 1 TO H1,13	04490001
M 1 TO H1,13	04500001
M 1 TO H1,13	04510001
M 1 TO H1,13	04520001
M 1 TO H1,13	04530001
M 1 TO H1,13	04540001
M 1 TO H1,13	04550001
M 1 TO H1,13	04560001
M 1 TO H1,13	04570001
M 1 TO H1,13	04580001
M 1 TO H1,13	04590001
M 1 TO H1,13	04600001
M 1 TO H1,13	04610001
M 1 TO H1,13	04620001
M 1 TO H1,13	04630001
M 1 TO H1,13	04640001
M 1 TO H1,13	04650001
M 1 TO H1,13	04660001
M 1 TO H1,13	04670001
M 1 TO H1,13	04680001
M 1 TO H1,13	04690001
M 1 TO H1,13	04700001
M 1 TO H1,13	04710001
M 1 TO H1,13	04720001
M 1 TO H1,13	04730001
M 1 TO H1,13	04740001

```

      ADD 203,6 TO H100,10
      MULTIPLY 116 BY 203,6 GIVING H140,6
      ADD H140,6 TO H120,10
ENDIF
* H160,10 SHIPCOST 2 DECIMAL PLACES
IF 203,6 GT '000070'
      ADD 2860 TO H160,10
ELSE
      ADD 496 TO H160,10
ENDIF
EXIT
*
* D2-WRITE-NSN PROB, AND RATES
D2
M H1 TO R1,13
IF H30,10 GT ZEROS
      DIVIDE H30,10 INTO H50,15
      ADD 5 TO H50,15
      M H59 TO R14,5
      DIVIDE H30,10 INTO H70,13
      ADD 5 TO H70,13
      M H77 TO R19,5
ELSE
      M ZEROS TO R14,5
      M ZEROS TO R19,5
ENDIF
IF H100,10 GT ZEROS
      DIVIDE H100,10 INTO H120,12
      ADD 5 TO H120,12
      M H125 TO R24,6
ELSE
      M '000082' TO R24,6
ENDIF
IF H200,10 GT ZEROS
      DIVIDE H200,10 INTO H160,13
      ADD 5 TO H160,13
      M H165 TO R30,7
ELSE
      M '0000910' TO R30,7
ENDIF
IF R24,6 EQ ZEROS
      M '000082' TO R24,6
ENDIF
IF R30,7 EQ ZEROS
      M '0000910' TO R30,7
ENDIF
W2
M ZEROS TO H14,500
EXIT
/*
//STEP5 EXEC PGM=IERRC000,REGION=990K,PARM='CORE=MAX'
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=*
//SORTMSG DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=GOR.BAKER.P0016NS,
// DISP=(SHR)
//SORTOUT DD DSN=GOR.BAKER.P0016SN,
// UNIT=WORKD,DISP=(NEW,PASS,DELETE),
// SPACE=(TRK,(13,91),RLSE),VOL=SER=WORKDS
//SORTWK01 DD UNIT=WORKD,SPACE=(TRK,913)
//SORTWK02 DD UNIT=WORKD,SPACE=(TRK,913)
//SORTWK03 DD UNIT=WORKD,SPACE=(TRK,913)
//SORTWK04 DD UNIT=WORKD,SPACE=(TRK,913)
//SORTWK05 DD UNIT=WORKD,SPACE=(TRK,913)
//SORTWK06 DD UNIT=WORKD,SPACE=(TRK,913)

```

```

02880001
02890001
02900001
02910001
02920001
02930001
02940001
02950001
02960001
02970001
02980001
02990001
03000001
03010001
03020001
03030001
03040001
03050001
03060001
03070001
03080001
03090001
03100001
03110001
03120001
03130001
03140001
03150001
03160001
03170001
03180001
03190001
03200001
03210001
03220001
03230001
03240001
03250001
03260001
03270001
03280001
03290001
03300001
03310001
03320001
03330001
03340001
03350001
03360001
03370001
03890002
03900001
03910001
03920001
03930001
03940001
03950001
03960001
03970001
03980001
03990001
04000001
04010001
04020001
04030001
04040001

```

GOBACK	02220001
*	02230001
* D1-ACCUMULATE QTY, WGT, AND COSTS	02240001
D1	02250001
* H20,1 SERVICE CODE	02260001
IF 147 EQ 'A' OR 'B' OR 'C' OR 'W'	02270001
M 'A' TO H20,1	02280001
ENDIF	02290001
IF 147 EQ 'N' OR 'P' OR 'Q' OR 'R' OR 'V' OR 'L' OR 'M' OR 'Z'	02300001
M 'N' TO H20,1	02310001
ENDIF	02320001
IF 147 EQ 'D' OR 'E' OR 'F' OR 'J'	02330001
M 'F' TO H20,1	02340001
ENDIF	02350001
* H30,10 MRO QTY	02360001
ADD 193,5 TO H30,10	02370001
* H50,10 PWR QTY	02380001
* H70,10 PWR DISPOSAL QTY WITH 2 DECIMAL PLACES	02390001
IF H20,1 EQ 'A'	02400001
IF 168,3 EQ '3AA' OR 'WR2'	02410001
ADD 193,5 TO H50,10	02420001
MULTIPLY 3 BY 193,5 GIVING H90,6	02430001
ADD H90,6 TO H70,10	02440001
ENDIF	02450001
IF H20,1 EQ 'N'	02460001
IF 168,1 EQ 'P'	02470001
ADD 193,5 TO H50,10	02480001
MULTIPLY 5 BY 193,5 GIVING H90,6	02490001
ADD H90,6 TO H70,10	02500001
ENDIF	02510001
IF H20,1 EQ 'N'	02520001
IF 168,3 EQ '714'	02530001
ADD 193,5 TO H50,10	02540001
ADD 193,5 TO H70,8	02550001
ENDIF	02560001
IF H20,1 EQ 'F'	02570001
IF 168,3 EQ '262' OR '596'	02580001
ADD 193,5 TO H50,10	02590001
ADD 193,5 TO H70,8	02600001
ENDIF	02610001
IF H20,1 EQ 'F'	02620001
IF 168,3 NE '262'	02630001
IF 168,3 NE '596'	02640001
IF 168,3 NE ' '	02650001
ADD 193,5 TO H50,10	02660001
MULTIPLY 25 BY 193,5 GIVING H90,6	02670001
ADD H90,6 TO H70,10	02680001
ENDIF	02690001
* H100,10 MROWGT	02700001
* H200,10 SHIPWGT	02710001
* H120,10 TRANCOST 2 DECIMAL PLACES	02720001
ADD 203,6 TO H200,10	02730001
IF 180,1 EQ 'A' OR 'B' OR 'D' OR 'J' OR 'K' OR 'N' OR 'Q' OR 'S' OR 'T'	02740001
ADD 203,6 TO H100,10	02750001
ADD 181,7 TO H120,10	02760001
ENDIF	02770001
IF 180,1 EQ 'U' OR 'V' OR '5'	02780001
ADD 203,6 TO H100,10	02790001
ADD 181,7 TO H120,10	02800001
ENDIF	02810001
IF 180,1 EQ 'G'	02820001
ADD 203,6 TO H100,10	02830001
MULTIPLY 24 BY 203,6 GIVING H140,6	02840001
ADD H140,6 TO H120,10	02850001
ENDIF	02860001
IF 180,1 EQ 'H'	02870001

```

//SYSIN DD *
SORT FIELDS=(1,13,CH,A)
END
/*
//STEP6 EXEC PGM=ATJ91U
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSUT1 DD DSN=GOR.BAKER.P8016SN,DISP=(SHR,CATLG)
//SYSUT1A DD DSN=GOR.BAKER.P8016AD,DISP=SHR
//SYSUT2 DD DSN=GOR.BAKER.P8016DN,
// UNIT=3380,DISP=(NEW,CATLG,DELETE),VOL=SER=WORKDS,
// DCB=(LRECL=169,BLKSIZE=23322,RECFM=FB),
// SPACE=(TRK,(100,100),RLSE)
//SYSIN DD *
NOREAD
READ
READA

* L1 MAINLINE
L1
IF 55,9 EQ ZEROS
M '000011280' TO N55,9
ENDIF
IF 49,6 EQ ZEROS
M '001604' TO N49,6
ENDIF
IF 43,6 EQ ZEROS
M '001334' TO N43,6
ENDIF
IF N1,13 EQ A1,13
MOVE 1 TO R1,146
MOVE A14 TO R147,5
MOVE A19 TO R152,5
MOVE A24 TO R157,6
MOVE A30 TO R163,7
W2
READ
GO 1
ENDIF
IF N1,13 LT A1,13
MOVE 1 TO R1,146
MOVE ZEROS TO R147,5
MOVE ZEROS TO R152,5
MOVE '000082' TO R157,6
MOVE '0000910' TO R163,7
W2
READ
GO 1
ENDIF
IF N1,13 GT A1,13 READA ENDIF
GO 1
/*
//STEP7 EXEC PGM=ATJ91U
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSUT1 DD DSN=GOR.BAKER.P8016DN,DISP=(SHR)
//SYSUT2 DD DSN=GOR.BAKER.P8016ND,
// SPACE=(TRK,(12,50),RLSE),DISP=(OLD,KEEP),
// UNIT=3380,VOL=SER=DORO02,DCB=(LRECL=163,RECFM=FB,BLKSIZE=23472)
//SYSIN DD *
M 1 TO 1,156
SUBTRACT 152,5 FROM 147,5 GIVING H1,5
M H1 TO 147,5
M 157 TO H157,6
ADD H157,7 TO 163,7 GIVING H10,7
M H10 TO 157,7
W2
/*
//

```

```

04050001
04060001
04070001
04080001
04090002
04100001
04110001
04120001
04130001
04140001
04150001
04160001
04170001
04180001
04190001
04200001
04210001
04220001
04230001
04240001
04250001
04260001
04270001
04280001
04290001
04300001
04310001
04320001
04330001
04340001
04350001
04360001
04370001
04380001
04390001
04400001
04410001
04420001
04430001
04440001
04450001
04460001
04470001
04480001
04490001
04500001
04510001
04520001
04530001
04540001
04550001
04560001
04880002
04890001
04900001
04910001
04920001
04930003
04940001
04950001
04960001
04970001
04980001
04990001
05000001
05010001
05020001
05030001
05040003

```

APPENDIX K

GOR.BAKER.JCI (P8016B) Code

```
//GOR6001C JOB (6001,GOR), 'BAKER',MSGCLASS=V,CLASS=6,NOTIFY=GOR6001
//STEP1 EXEC PGM=SELCOP,PARM=S
//SYSUDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUT1 DD DSN=SAN.DATA.LJAC7801,DISP=SHR,
// DCB=(LRECL=144,BLKSIZE=23328,RECFM=FB,DEN=3),
// LABEL=(1,SL,,EXPDT=980000),UNIT=TAPE,
// VOL=SER=(016278,015936,027885,016871,016699,016378,016444,016367,
// 016359,016340,016356,002203)
//SYSUT2 DD DSN=GOR.BAKER.P8016B0,
// DISP=(NEW,CATLG,DELETE),UNIT=TAPE,LABEL=RETPD=99,VOL=(,,12)
//SYSIN DD *
RANGE 0000001 9999999
/*
//STEP2 EXEC PGM=ATJ91U,REGION=990K
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUT1 DD DSN=GOR.BAKER.P8016B0,DISP=SHR,
// DCB=(LRECL=144,BLKSIZE=23328,RECFM=FB,DEN=3),
// LABEL=(1,SL,,EXPDT=980000),UNIT=TAPE
//SYSUT2 DD DSN=GOR.BAKER.P8016B,DISP=(NEW,CATLG,DELETE),
// DCB=(LRECL=120,BLKSIZE=23400,RECFM=FB,DEN=4),
// LABEL=RETPD=180,UNIT=TAPE,
// VOL=(,,12)
//SORTPARM DD *
MSG=AP
//SORTWK01 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK02 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK03 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK04 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK05 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK06 DD UNIT=WORKD,SPACE=(TRK,912)
//SYSIN DD *
SORT FIELDS=(7,1,CH,A)
RECORD LRECL=120
PHASE INPUT
IF 31,13 NUMERIC
M 25 TO 1,120
RELEASE
ENDIF
PHASE OUTPUT
MOVE 1 TO 1,120
W2
/*
//STEP3 EXEC PGM=ATJ91U,REGION=990K
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUT1 DD DSN=GOR.BAKER.P8016B,DISP=SHR
//SYSUT2 DD DSN=GOR.BAKER.P8016BS,DISP=(NEW,CATLG,DELETE),
// DCB=(LRECL=19,BLKSIZE=23465,RECFM=FB),
// LABEL=RETPD=180,UNIT=TAPE,
// VOL=(,,10)
//SORTPARM DD *
MSG=AP
//SORTWK01 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK02 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK03 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK04 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK05 DD UNIT=WORKD,SPACE=(TRK,912)
//SORTWK06 DD UNIT=WORKD,SPACE=(TRK,912)
//SYSIN DD *
SORT FIELDS=(1,13,CH,A)
RECORD LRECL=19
PHASE INPUT
```

M 'G' TO 19,1	00019720
IF 47,7 EQ 'QTY ADJ'	00019820
IF 39,2 EQ 'LL'	00019920
M 'L' TO 19,1	00020020
ENDIF	00020120
IF 68,5 LT '99999'	00020220
IF 36,3 EQ 'XR1' OR 'XR3'	00020320
M 1 TO 1,13	00020420
M 68 TO 14,5	00020520
RELEASE	00020620
ENDIF	00020720
IF 68,5 LT '99999'	00020820
IF 36,3 EQ 'XA1'	00020920
IF 47,7 EQ 'QTY ADJ'	00021020
M 1 TO 1,13	00021120
M 68 TO 14,5	00021220
RELEASE	00021320
ENDIF	00021420
PHASE OUTPUT	00021520
MOVE 1 TO 1,19	00021620
W2	00021720
/*	00021820
//STEP4 EXEC PGM=ATJ91U	00021921
//SYSPRINT DD SYSOUT=*	00022020
//SYSUT1 DD DSN=GOR.BAKER.P8016BS,DISP=SHR	00022120
//SYSUT2 DD DSN=GOR.BAKER.P8016BA,DISP=(NEW,CATLG,DELETE),	00022220
UNIT=3380,VOL=SER=WORKWS,SPACE=(TRK,(30,100),RLSE),	00022320
UNIT=TAPE,LABEL=RETPD=180,VOL=(,,,9),	00022422
DCB=(LRECL=23,BLKSIZE=23460,RECFM=FB)	00022520
//SYSIN DD *	00022620
* MAINLINE	00022720
IF EOJ	00022820
DO 2 EOJ ENDIF	00022920
IF SYSUT1 = 1	00023020
M 1 TO H1,13 ENDIF	00023120
IF 1,13 NE H1,13	00023220
DO 2	00023320
M 1 TO H1,13 ENDIF	00023420
DO 1	00023520
GOBACK	00023620
*	00023720
* D1-ACCUMULATE DISPOSAL QTY BY NSN	00023820
*	00023920
D1	00024020
*	00024120
IF 14,5 NUMERIC	00024220
M 14 TO H14,5	00024320
ELSE	00024420
M ZEROS TO H14,5	00024520
ENDIF	00024620
IF 19,1 EQ 'L'	00024720
SUBTRACT H14,5 FROM H100,10	00024820
ELSE	00024920
ADD H14,5 TO H100,10	00025020
ENDIF	00025120
EXIT	00025220
*	00025320
* D2-WRITE-NSN AND DISPOSAL QTY	00025420
*	00025520
D2	00025620
IF H100,10 LT ZEROS	00025720
M ZEROS TO H100,10	00025820
	00025920
	00026020
	00026120
	00026220

ENDIF	00026320
M H1 TO R1,13	00026420
M H100 TO R14,10	00026520
W2	00026620
M ZEROS TO H100,10	00026720
EXIT	00026820
/*	00026920
//STEP5 EXEC PGM=ATJ91U	00027021
//SYSPRINT DD SYSOUT=*	00027120
//SYSUDUMP DD SYSOUT=*	00027220
//SYSUT1 DD DSN=GOR.BAKER.P8016ND,DISP=SHR	00027320
//SYSUT1A DD DSN=GOR.BAKER.P8016BA,DISP=SHR	00027420
//SYSUT2 DD DSN=GOR.BAKER.P8016NB,	00027520
// UNIT=3380,DISP=(NEW,CATLG,DELETE),VOL=SER=WORKWS,	00027620
// DCB=(LRECL=173,BLKSIZE=23355,RECFM=FB),	00027720
// SPACE=(TRK,(27,100),RLSE)	00027820
//SYSIN DD *	00027920
NOREAD	00028020
READ	00028120
READA	00028220
	00028320
* L1 MAINLINE	00028420
L1	00028520
IF EOJ	00028620
P 'NUMBER OF NSNS WITHOUT DISPOSALS ='	00028720
P H100,10	00028820
EOJ	00028920
IF N1,13 EQ A1,13	00029020
MOVE 1 TO R1,100	00029120
MOVE A14 TO R164,10	00029220
W2	00029320
READ	00029420
GO 1	00029520
ENDIF	00029620
IF N1,13 LT A1,13	00029720
MOVE 1 TO R1,100	00029820
MOVE ZEROS TO R164,10	00029920
W2	00030020
READ	00030120
ADD 1 TO H100,10	00030220
GO 1	00030320
ENDIF	00030420
IF N1,13 GT A1,13 READA ENDIF	00030520
GO 1	00030620
/*	00030720
//STEP6 EXEC FORIVCLG,REGION=2000K	00030821
//FORT.SYSIN DD *	00030920
INTEGER N, IQFD, IADQ	00031020
DOUBLE PRECISION ADQ, QFD, DISPQ, DISPR, PDISP, UPRICE, MADQ	00031120
REAL NPWR, UCUBE, UWT, SLM, RPWR, TRANH	00031220
CHARACTER*42 A1	00031320
CHARACTER*18 A2	00031420
CHARACTER*9 A3	00031520
CHARACTER*17 A4	00031620
CHARACTER*9 A5	00031720
C LOOP THRU NSNS	00031820
DO 900 N=1,99999	00031920
READ(10,10,END=920) A1,UCUBE,UWT,UPRICE,SLM,A2,MADQ,A3,	00032020
+ ADQ,A4,QFD,A5,RPWR,NPWR,TRANH,DISPQ	00032120
10 FORMAT(A42,F6.3,F6.2,F9.2,F3.1,A18,F9.1,A9,F9.0,A17,F9.0,A9,	00032220
+ F5.4,F5.4,F7.4,F10.0)	00032320
IF(ADQ.EQ.0.0)THEN	00032420
ADQ=QFD*4.0	00032520
END IF	00032620
IF(ADQ.GT.0.0)THEN	00032720
DISPR=2.0*DISPQ/ADQ	00032820

END IF	00032920
IF(ADQ.EQ.0.0.AND.DISPR.GT.0.0)THEN	00033020
DISPR=1.0	00033120
END IF	00033220
IF(ADQ.EQ.0.0.AND.DISPR.LT.1.0)THEN	00033320
DISPR=0.0	00033420
END IF	00033520
IF(DISPR.GT.1.0)THEN	00033620
DISPR=1.0	00033720
END IF	00033820
IF(DISPR.LT.0.0)THEN	00033920
DISPR=0.0	00034020
END IF	00034120
IF(DISPR.GT.NPWR)THEN	00034220
PDISP=DISPR-NPWR	00034320
ELSE	00034420
PDISP=0.0	00034520
END IF	00034620
IF(PDISP.GT.1.0)THEN	00034720
PDISP=1.0	00034820
END IF	00034920
IF(PDISP.LT.0.0)THEN	00035020
PDISP=0.0	00035120
END IF	00035220
IQFD=INT(QFD)	00035320
IADQ=INT(ADQ)	00035420
WRITE(8,30)	00035520
+ A1,UCUBE,UWT,UPRICE,SLM,A2,MADQ,A3,	00035620
+ IADQ,A4,IQFD,A5,RPWR,NPWR,TRANH,PDISP	00035720
30 FORMAT(A42,F7.3,F7.2,F10.2,F4.1,A18,F10.1,A9,I9,A17,I9,A9,	00035820
+ F6.4,F6.4,F8.4,F8.6)	00035920
900 CONTINUE	00036020
920 STOP	00036120
END	00036220
/*	00036320
//GO.FT10F001 DD DSN=GOR.BAKER.P8016NB,DISP=SHR	00036420
//GO.FT08F001 DD DSN=GOR.BAKER.P8016PD,DISP=(NEW,CATLG,DELETE),	00036520
// UNIT=3380,SPACE=(TRK,(20,29),RLSE),VOL=SER=WORKWS,	00036622
// DCB=(RECFM=FB,LRECL=179,BLKSIZE=23449)	00036720
/*	00036820
//STEP7 EXEC SPSSX,CYLS='10,180'	00036921
TITLE 'GOR.BAKER.P8016PD FREQUENCY REPORT'	00037020
DATA LIST FILE = INFILE /	00037120
LDISP 35-39	00037220
SHM 40-41	00037320
SLM 67-70	00037420
BOQ 71-79	00037520
IAQ 80-88	00037620
MADQ 89-98	00037720
OWRMRP 99-107	00037820
ADQ 108-116	00037920
ADF 117-123	00038020
PCM 124-126	00038120
ROP 127-133	00038220
QFD 134-142	00038320
QFR 143-151	00038420
RPWR 152-157	00038520
NPWR 158-163	00038620
TRANH 164-171	00038720
PDISP 172-179	00038820
SUBTITLE 'PEACETIME DISPOSAL PROB STEPWISE REGRESSION'	00038920
REGRESSION VARIABLES= LDISP TO PDISP	00039020
/CRITERIA=PIN(.01) POUT(.25) TOLERANCE(.01)	00039120
/DEP=PDISP /STEPWISE	00039220
/PARTIALPLOT	00039320
FINISH	00039420

//INFILE DD DSN=GOR.BAKER.P8016PD,	00039520
// DISP=SHR	00039620
//*	00039721
//STEP8 EXEC PGM=ATJ91U	00039821
//SYSPRINT DD SYSOUT=*	00039917
//SYSUDUMP DD SYSOUT=*	00040017
//SYSUT1 DD DSN=GOR.BAKER.P8016PD,DISP=(SHR)	00040117
//SYSUT2 DD DSN=GOR.BAKER.P8016DB,	00040217
// SPACE=(TRK,(6,10),RLSE),DISP=(OLD,KEEP),	00040322
// UNIT=380,VOL=SER=DOR002,DCB=(LRECL=77,BLKSIZE=23408,RECFM=FB)	00040417
//SYSIN DD *	00040517
M 1 TO 1,13	00040617
M 14 TO 14,2	00040717
M 16 TO 16,19	00040817
M 40 TO 35,2	00040917
M 50 TO 37,7	00041017
M 67 TO 44,4	00041117
M 117 TO 48,7	00041217
M 124 TO 55,3	00041317
M 152 TO 58,20	00041417
W2	00041519
/*	00041618
//	00041718

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE May 1989		3. REPORT TYPE AND DATES COVERED Final	
4. TITLE AND SUBTITLE Medical Acquisition Shelf-Life System (MASS) Decision Support Model Systems Documentation				5. FUNDING NUMBERS	
6. AUTHOR(S) Mr. Richard E. Baker Mr. Rick L. Jernigan					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) HQ Defense Logistics Agency Operations Research and Economic Analysis Office (DLA-LO) Cameron Station Alexandria, VA 22304-6100				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Defense Logistics Agency Cameron Station Alexandria, VA 22304-6100				10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES					
12a. DISTRIBUTION/AVAILABILITY STATEMENT Public Release; Unlimited Distribution				12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The Medical Acquisition Shelf-Life (MASS) Model is a decision aid to assist procurement analysts in evaluating alternative bids for stocked medical shelf-life items. MASS attempts to identify the best value bid by balancing longer shelf-life against higher purchase price in order to identify the bid with the lowest life cycle costs. The objectives of the systems documentation are to briefly review the model features, document the MASS programs, describe the MASS files, and explain the procedures for updating the MASS data. The objectives of the user's guide are to describe the model's features, instruct the user in using MASS, and explain the rationale of the model to vendors.					
14. SUBJECT TERMS Medical, Shelf-Life, Life-Cycle Costs				15. NUMBER OF PAGES 41	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT		