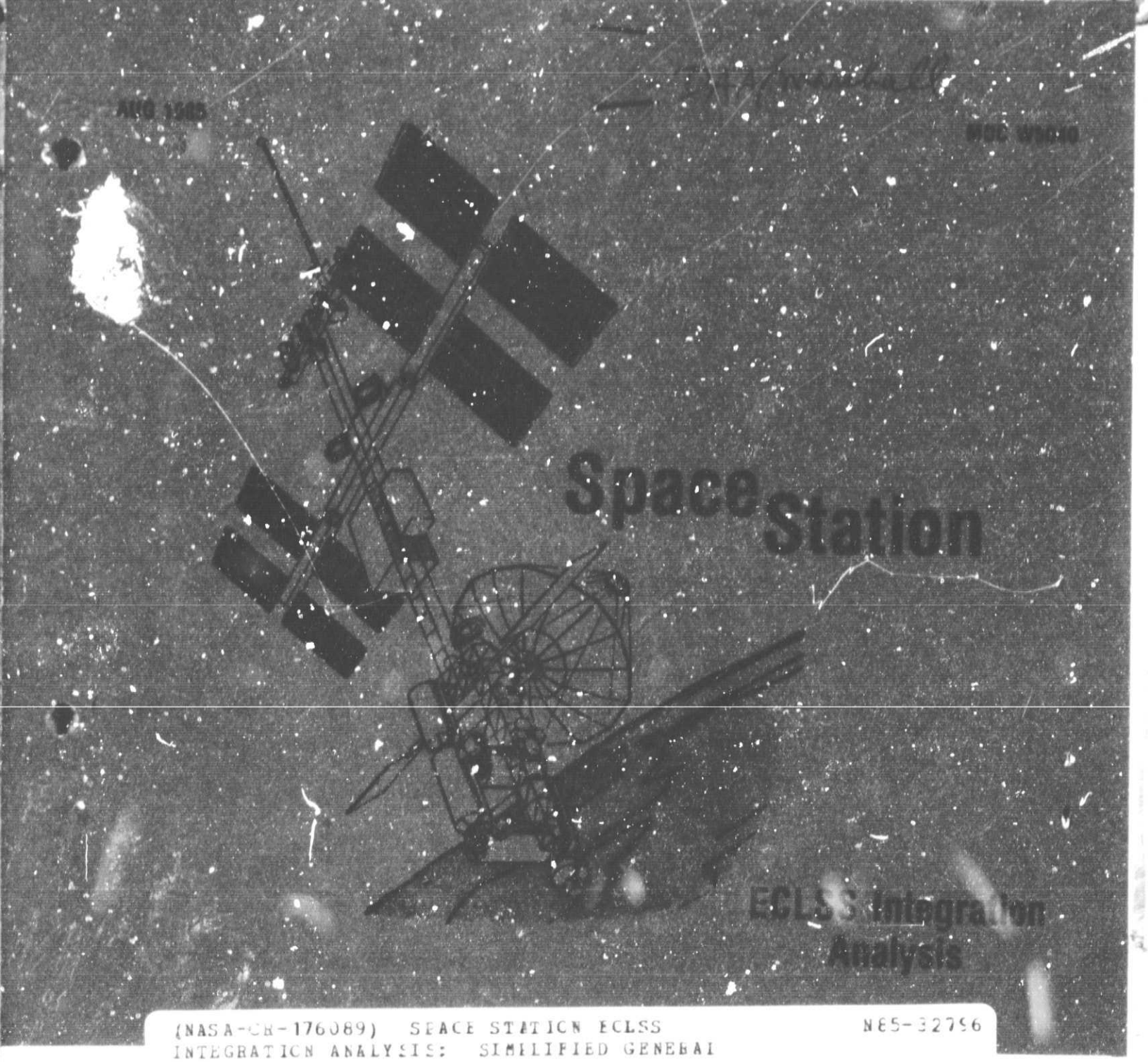

MDTSCO
HUNTSVILLE OPERATIONS

P.O. Box 1181, Huntsville, Alabama 35807 (205) 453-5077

Simplified General Cluster Systems Model
ECLS System Assessment Program Enhancements

MCDONNELL DOUGLAS
CORPORATION

Space Station

ECLSS Integration Analysis

Simplified General Cluster Systems Model
ECLS System Assessment Program Enhancements

PREPARED BY: R E Ferguson
R. E. FERGUSON
CHIEF, ANALYSIS &
DESIGN ENGINEERING

APPROVED BY: E M Chewning
E. M. CHEWNING
MANAGER, SPACE STATION
PROJECTS

PREPARED FOR THE NASA MARSHALL SPACE FLIGHT CENTER UNDER CONTRACT NO. NAS8-36407,
EFFECTIVE DATE: 9 MAY 1985 (SOW TASK b, MODEL 1, WBS 1.2.1)

MCDONNELL DOUGLAS TECHNICAL SERVICES COMPANY, INC.
HUNTSVILLE OPERATIONS
P.O. BOX 1181
HUNTSVILLE, AL 35807

TABLE OF CONTENTS

SECTION 1
INTRODUCTION

This report was prepared by McDonnell Douglas Technical Services Company (MDTSCO) to document the data base verification of the ECLS Systems Assessment Program (ESAP) and changes made to enhance the flexibility of the water recovery subsystem simulations. This report describes all changes which were made to the data base values and the software enhancements performed. The refined model documented herein constitutes the MDTSCO submittal of the General Cluster Systems Model described under SOW paragraph b. A source listing of the current version of ESAP is provided in Appendix A.

SECTION 2
DISCUSSION

The ECLS System Assessment Program is a MSFC developed program which is used as a conceptual system design/trade study tool. This program was provided to MDTSCO to form the basis for the Space Station General Cluster Systems Model. The tasks performed by MDTSCO include (1) verification of the data base values and component simulations, and (2) software enhancements to increase the flexibility of the program in the water recovery subsystem simulation.

The verification of ESAP included assessment of the accuracy of the data base values i.e. the weight, power, and volume data and verification of the fundamental physics of the component simulations. The data based values for each subsystem were compared to current values found in recent literature and documentation. Changes were considered necessary in the data based values for the subsystems used in the water reclamation system. A list of the changes that were made are provided in Table 1.

All other data based values are in agreement with current documented values. The verification of the component simulations indicated that the stoichiometric relationships were correct in all of the component simulations.

The principal software enhancement consists of incorporating operational changes to allow the engineer to address inter-mixing of flow streams between the same and/or different water recovery systems. Addition of this greater flexibility to ESAP necessitated a major rewrite to the water reclamation input scheme. The objective is to enable the user to place subsystems in series and parallel. The solution to this problem was to define the type of water to be recovered first and then define the path that the primary stream followed. After the primary flow path for a particular water group has been established, the quality of the water recovered is defined. With all water groups, their corresponding flow paths, and water quality defined, secondary flow paths are established between existing subsystems and new subsystems.

In order to achieve the desired results, several variables and arrays were added to the input section of program. Below is a brief description of the variables and arrays that were added to the main program:

JFLOW(I,J)	This array contains the primary flow path. 'I' represents the water group number and 'J' represents the order of the subsystem selected.
KFLOW(I,J)	This array contains the secondary flow paths to predefined subsystems. 'I' represents the water group number that the flow is coming from and 'J' represents the subsystem unit number that the water is coming from. The value this array receives is a compound number 'KL', where the first digit 'K' represents the water group number the flow is going to and the second digit 'L' represents the subsystem unit number the water is entering.

LFLOW(I,J)	This array contains the secondary flow paths to new subsystems. 'I' represents the flow group the water is coming from and 'J' represents the subsystem unit number. This array receives the value of the subsystem number the water is entering.
IUNIT	Variable counter that keeps track of the number of water reclamation subsystems chosen.
GROUP	Variable counter that keeps track of the number of water groups selected.
ANSWER	Alphanumeric character that is used in reading answers to questions.

The program uses an iterative approach in evaluating the water reclamation system. The total throughput of water for each flow group is determined first, with each flow group subsequently put through its respective flow path. The flow of water is modified each time it entered a subsystem by the efficiency of that subsystem. The flow of water rejected from each subsystem is also calculated.

After each primary flow stream is evaluated, the program then computes the secondary flow streams from the designated subsystems. This is accomplished by sending the desired secondary stream into the designated subsystem and evaluating this stream as if it were a primary flow. If the subsystem is a predefined subsystem then the secondary water stream is processed through the remaining subsystems in the new flow path and the quality of the water will be the same as that of the flow path entered.

Several new arrays and variables were added to the WRS subroutine, they are:

E(I,J)	This array receives the subsystem efficiency. This array was increased to a two dimensional array in order to accommodate the subsystem selection method. 'I' represents the flow group number and 'J' represents the subsystem number.
EFF(I)	Array that contains the value of the subsystem efficiency, where 'I' represents the overall subsystem number.
SSWATER(I)	Array that contains the amount of secondary water being directed from a new unit into a potable tank. 'I' is the flow group number.
TSWATER(I)	This array receives the amount of secondary water being directed from a new unit into a hygiene tank. 'I' is the flow group number.

TWATER(I)	This array is assigned the value of the amount of water of each water group. This is used in tracking the different water types.
SWATER(I,J)	Array that contains the amount of secondary water from each subsystem. 'I' represents the flow group and 'J' represents the subsystem number.
PSWATER(I)	This array receives the final amount of processed secondary water. 'I' represents the flow group that the secondary water stream entered.
ZCONDR ZREDH2OR ZHYGIENR ZURINR ZDISHWR ZCLOTHR	These variables are used in computing the amount of each water type that was recovered.
HYGREC	Variable used in calculating the amount of hygiene water recovered.

In addition to the above major modifications a few minor changes were made to other subroutines. The value for the total load on each water reclamation subsystem was taken out of subroutine ANALYZE and put into the WRS subroutine, to facilitate computations. The auxiliary equipment equation in subroutine RESUP was changed in order to accommodate the method used to compute the individual subsystem loads. Additions were also made to subroutine INITL to initialize the new arrays.

The operation of the program continues to follow the menu scheme present in the original program received from NASA. The menu functions to support the enhancements to the WRS simulation have been implemented to solicit the necessary data from the user.

TABLE 1.
MODIFIED ESAP DATA BASE PARAMETERS

COMPONENT	CHANGES	REFERENCES
VCD	Initial volume changed from 17.0 to 25.6 cubic feet.	1
TIMES	Initial volume changed from 13.0 to 20.0 cubic feet.	1
AIR EVAP	Initial volume changed from 10.0 to 15.0 cubic feet.	1
MULTIFILTRATION	Average power required on night side changed from 0.0 to 30.0 watts.	1
HYPERFILTRATION	Average power required to daylight and night side changed from 0.0 to 272.0 watts. Initial weight changed from 0.0 to 460.0 pounds. Initial volume changed from 0.0 to 30.8 cubic feet.	2
REVERSE OSMOSIS	Average power required on daylight and night side changed from 0.0 to 45.0 watts. Initial weight changed from 0.0 to 125.0 pounds. Initial volume changed from 0.0 to 30.8 cubic feet.	3

REFERENCES

1 Space Station Regenerative Life Support Equipment Weight, Power, and Volume Data, NASA-MSFC EP45/85-34, Mr. Frederick, April 2, 1985.

2 Hyperfiltration Wash Water Recovery Subsystem Design and Test Results, SAE 83112, R. P. Reysa, Boeing Aerospace, July 1983.

3 Membrane Based Water and Energy Recovery Systems for Manned Space Station, SAE 85B45, Rob Ray, Bend Research, July 1985.

SECTION 3
SUMMARY

The verification process resulted in relatively minor changes to the ESAP data base as was expected. The changes to the water recovery system software to address intermixing of the flow streams required a major rewrite, but the resulting software allows the engineer the flexibility to address hybrid systems. In accordance with the SOW, future changes to ESAP will include consideration of adding the capability to address module heat balances and updates to the ECLS equipment algorithms.

APPENDIX A

PROGRAM LISTING

```
      PROGRAM ESAP2
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP,TANKED
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),
     * COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      COMMON /FILE/ FNAME(NSYS)
      COMMON /IWRT/ IWRITE
      CHARACTER*12 FNAMF
      CHARACTER*6 ADUMP
      CHARACTER*7 AVAL
      CHARACTER*1 ANSWER
      CHARACTER*15 SSNAME
      CHARACTER*6 TANKED
      OPEN(UNIT=1,FILE='ESAPDATA.DAT',STATUS='UNKNOWN')
C
C        SUBROUTINE DIRECTORY
C
C    SUBROUTINE          DESCRIPTION
C
C    ESAP              MAIN ROUTINE, SELECT SUBSYSTEMS, CONTROLS PROGRAM FLOW
C    ANALIZE           PRIMARY EXECUTION ROUTINE, CONTROLS COMPUTATION FLOW,
C                       COMPUTES TOTALS
C    ARS               COMPUTATION ROUTINE FOR AIR REVITALIZATION SUBSYSTEMS
C    WRS               COMPUTATION ROUTINE FOR WATER RECLAMATION SUBSYSTEMS
C    SIZE              ROUTINE CALLED BY ALL SUBSYSTEM SIZING ROUTINES TO SIZE
C                       SUBSYSTEM BASED ON LOAD, SIZING RATIOS
C    SBRO2GEN          COMPUTES OXYGEN REQUIREMENTS
C    SFWES             SUBSYSTEM SIZING ROUTINE
C    SPOLLF            SUBSYSTEM SIZING ROUTINE
C    SPOLSF            SUBSYSTEM SIZING ROUTINE
C    O2STOR            SUBSYSTEM SIZING ROUTINE
C    EDC               SUBSYSTEM SIZING ROUTINE
C    SAWD              SUBSYSTEM SIZING ROUTINE
C    HMOLSV            SUBSYSTEM SIZING ROUTINE
C    LIOH              SUBSYSTEM SIZING ROUTINE
C    FBDMSV            SUBSYSTEM SIZING ROUTINE
C    SKLMSV            SUBSYSTEM SIZING ROUTINE
C    BOSCH             SUBSYSTEM SIZING ROUTINE
C    SABAT             SUBSYSTEM SIZING ROUTINE
C    SABCH4            SUBSYSTEM SIZING ROUTINE
C    VCD               SUBSYSTEM SIZING ROUTINE
C    TIMES             SUBSYSTEM SIZING ROUTINE
C    AIREVP            SUBSYSTEM SIZING ROUTINE
C    VPCAR             SUBSYSTEM SIZING ROUTINE
C    MF                SUBSYSTEM SIZING ROUTINE
C    HF                SUBSYSTEM SIZING ROUTINE
C    RO                SUBSYSTEM SIZING ROUTINE
```

```
C   SRSSUB          WRITES SUMMARY DATA TO SCREEN/FILE (FIRST 6 SCREENS OF DATA)
C   WARSMAT         WRITES ARS MATERIALS SUMMARY TO SCREEN/FILE
C   WWRSMAT         WRITES WRS MATERIALS SUMMARY TO SCREEN/FILE
C   RESUP           CALCULATES AUX EQUIP REQD BY SYSTEM:INCL H2O,TANKS,MATL LOST
C   WRESUP          WRITES AUX EQUIP REQUIREMENTS TO SCREEN (SCREENS 9 & 10)
C   DESGNS          CONTAINS DESIGN LOADS DATABASE
C   SSDATS          CONTAINS SUBSYSTEM DATABASE
C   AUXILS          CONTAINS AUXILIARY EQUIPMENT DATABASE
C   SBRSGN          WRITES DESIGN LOADS DATABASE TO SCREEN FOR CHANGE/REVIEW
C   DSGNIN          ALLOWS CHANGES TO DESIGN LOADS DATABASE
C   SSDTIN          ALLOWS CHANGES TO SUBSYSTEM DATABASE
C   SBRSSDAT        WRITES SUBSYSTEM DATABASE TO SCREEN FOR CHANGE/REVIEW
C   SBRAUXIL        WRITES AUXILIARY EQUIP DATABASE TO SCREEN FOR CHANGE/REVIEW
C   AUXILIN         ALLOWS CHANGES TO AUXILIARY EQUIPMENT DATABASE
C   AUXLSTRT        TAKES GENERIC AUX EQUIP DATABASE & DISTRIBUTES EQUIP FOR
C                    SUBSYSTEMS CHOSEN - PRIMARY EFFECT IS ON WRS
C   NUMUNIT         ALLOWS SPECIFICATION OF NUMBER AND RELATIVE SIZE OF
C                    SUBSYTEMS & REDUNDANT UNITS
C   STORE           STORES SUMMARY DATA FOR A GIVEN SYSTEM FOR LATER COMPARISON
C                    TO OTHER SYSTEMS
C   COMPARE         COMPARES UP TO 3 SYSTEMS BASED ON A POINT SYSTEM
C   POINTASG        COMPUTES POINT TOTALS OF EACH SYSTEM AND DISPLAYS RESULTS
C   POINTIN         ALLOWS SPECIFICATION OF POINTS FOR EACH SUMMARY ITEM
C   INITL           INITIALIZES ARRAYS FOR START OF NEW SYSTEM SELECTION
C
C
C  ITYPE(I,1) = SYSTEM NUMBER (1-50)
C  ITYPE(I,2) = POTABLE VS HYGIENE H2O RECOVERED(WRS),DUMP VS STORE(CO2RED)
C  ITYPE(I,3) = NUMBER OF UNITS (ITYPE(I,3)=2,N=6 - 2 3-MAN SYSTEMS)
C  ITYPE(I,4) = NUMBER OF BACKUP UNITS
C
      ITRACK=0
    5 CALL DESGNS
      CALL SSDATS
      CALL AUXILS
    4 CALL INITL
      ADUMP='STORED'
      NUMBER=0
      NUMB=0
      PRINT 6
    6 FORMAT(16(/))
      PRINT*,'    ECLS Systems Assessment Program (ESAP)'
      PRINT 3
      PRINT*,'DO YOU WANT TO WRITE OUTPUT DATA TO:'
      PRINT*,'  1. FILE ONLY'
      PRINT*,'  2. SCREEN ONLY'
      PRINT*,'  3. BOTH FILE AND SCREEN (DEFAULT)'
      READ(5,1) IWRITE
    1 FORMAT(I2)
      IF (IWRITE.LT.1.OR.IWRITE.GT.2) IWRITE=3
      PRINT 3
    3 FORMAT(5(/))
      PRINT*,'  AIR REVITALIZATION'
      PRINT*,'     '
      PRINT*,'O2 GENERATION'
      PRINT*,'   0. NONE'
      PRINT*,'  21. STATIC FEED - WES'
      PRINT*,'  22. SOLID POLYMER - LIQUID FEED'
      PRINT*,'  23. SOLID POLYMER - STATIC FEED'
      PRINT*,'  24. O2/H2 STORAGE'
```

```
30 PRINT*,'CHOOSE NUMBER OF SUBSYSTEM TO ANALIZE'
   READ(5,*,ERR=30) INUM
   IF (INUM.EQ.0) GOTO 25
   IF (INUM.LT.21.OR.INUM.GT.24) GOTO 30
   IF (INUM.EQ.24) THEN
     PRINT*,' SUBSYSTEM IS NOT CURRENTLY MODELLED'
     GOTO 30
   ENDIF
   NUMBER=NUMBER+1
   ITYPE(NUMBER,1)=INUM
   CALL NUMUNIT(ITYPE(NUMBER,3),ITYPE(NUMBER,4))
25 PRINT*,'     '
   PRINT*,'CO2 REMOVAL'
   PRINT*,'    0. NONE'
   PRINT*,'    1. EDC'
   PRINT*,'    2. SAWD'
   PRINT*,'    3. HYDROPHOBIC MOLE SIEVE'
   PRINT*,'    4. LIOH'
   PRINT*,'    5. FOUR BED MOLE SIEVE'
   PRINT*,'    6. SKYLAB MOLE SIEVE'
10 PRINT*,'CHOOSE NUMBER OF SUBSYSTEM TO ANALIZE'
   READ(5,*,ERR=10) INUM
   IF (INUM.EQ.0) GOTO 15
   IF (INUM.LT.1.OR.INUM.GT.6) GOTO 10
   IF (INUM.EQ.5.OR.INUM.EQ.6) THEN
     PRINT*,' SUBSYSTEM IS NOT CURRENTLY MODELLED'
     GOTO 10
   ENDIF
   NUMBER=NUMBER+1
   ITYPE(NUMBER,1)=INUM
   CALL NUMUNIT(ITYPE(NUMBER,3),ITYPE(NUMBER,4))
   IF (INUM.EQ.4.OR.INUM.EQ.6) GOTO 35
15 PRINT*,'     '
   PRINT*,'CO2 REDUCTION'
   PRINT*,'    0. NONE'
   PRINT*,'   11. BOSCH'
   PRINT*,'   12. SABATIER'
   PRINT*,'   13. SABATIER/CH4 DISSOCIATION'
20 PRINT*,'CHOOSE NUMBER OF SUBSYSTEM TO ANALIZE'
   READ(5,*,ERR=20) INUM
   IF (INUM.EQ.0) GOTO 35
   IF (INUM.LT.11.OR.INUM.GT.13) GOTO 20
   IF (INUM.EQ.13) THEN
     PRINT*,' SUBSYSTEM IS NOT CURRENTLY MODELLED'
     GOTO 20
   ENDIF
   NUMBER=NUMBER+1
   ITYPE(NUMBER,1)=INUM
   CALL NUMUNIT(ITYPE(NUMBER,3),ITYPE(NUMBER,4))
   IF (INUM.EQ.11.OR.INUM.EQ.12) THEN
     IF (INUM.EQ.11) AVAL='H2'
     IF (INUM.EQ.12) AVAL='CO2/CH4'
     PRINT 21
21   FORMAT(/,1X,'CHOOSE OPTION:')
     PRINT 22,AVAL
22   FORMAT(T4,'1. STORE ',A7)
     PRINT 23,AVAL
23   FORMAT(T4,'2. DUMP ',A7)
24   READ(5,*,ERR=24) INUM1
     IF (INUM1.NE.1.AND.INUM1.NE.2) GOTO 24
```

```
      NUMB=NUMBER
      ITYPE(NUMB,2)=INUM1
      IF (INUM1.EQ.2) ADUMP='DUMMED'
    ENDIF
    IUNIT=0
    GROUP=0
 35 DO 38 I=1,NWNUM
 38 IF (IWATER(I).EQ.0) GOTO 39
    GOTO 37
 39 PRINT*,'    '
    GROUP=GROUP+1
    PRINT*,'WATER RECLAMATION'
    PRINT 41
 41 FORMAT(/,1X,'ENTER NUMBER(S)-ONE PER LINE-OF WATER TO BE ',
   * 'RECOVERED IN THIS FLOW PATH')
    PRINT*,'SELECTION'
    PRINT*,'    '
    PRINT*,'    0. END SELECTION OF TYPE OF WATER RECOVERED'
    IF (IWATER(1).EQ.0) PRINT*,'    1. CONDENSATE WATER'
    IF (IWATER(2).EQ.0) PRINT*,'    2. CO2 REDUCTION'
    IF (IWATER(3).EQ.0) PRINT*,'    3. HYGIENE WATER'
    IF (IWATER(4).EQ.0) PRINT*,'    4. CLOTHES WASH WATER'
    IF (IWATER(5).EQ.0) PRINT*,'    5. URINE WATER'
    IF (IWATER(6).EQ.0) PRINT*,'    6. DISHWASHER WATER'
 45 READ(5,*,ERR=45) INUM1
    IF (INUM1.EQ.0) GOTO 31
    IF (INUM1.LT.0.OR.INUM1.GT.6) GOTO 45
    IF (IWATER(INUM1).NE.0) THEN
      PRINT*,' WATER TYPE HAS ALREADY BEEN SELECTED'
    ELSE
      IWATER(INUM1)=GROUP
    ENDIF
    GOTO 45
 31 PRINT*,'    '
 33 PRINT*,'    '
    PRINT*,'    '
    PRINT*,'DEFINE THE COMPLETE SUBSYSTEM ORDER FOR THE PRIMARY ',
   *       'FLOW STREAM OF THE CURRENT WATER GROUP'
    PRINT*,'    '
    PRINT*,'ENTER THE NUMBER OF THE SUBSYSTEM TO ',
   *       'RECEIVE WATER'
    PRINT*,'    '
    PRINT*,'    0. END WATER RECLAMATION SUBSYSTEM SELECTION'
    PRINT*,'   31. VCD'
    PRINT*,'   32. TIMES'
    PRINT*,'   33. AIR EVAP'
    PRINT*,'   34. VPCAR'
    PRINT*,'   41. MULTIFILTRATION'
    PRINT*,'   42. HYPERFILTRATION'
    PRINT*,'   43. REVERSE OSMOSIS'
 40 READ(5,*,ERR=40) INUM
    IF (INUM.EQ.0) GOTO 36
    IF (INUM.LT.31.OR.INUM.GT.43) GOTO 40
    IF (INUM.GT.34.AND.INUM.LT.41) GOTO 40
    IF (INUM.EQ.34) THEN
      PRINT*,' SUBSYSTEM IS NOT CURRENTLY MODELLED'
      GOTO 40
    ENDIF
    IUNIT=IUNIT+1
    NUMBER=NUMBER+1
```

```
      ITYPE(NUMBER,1)=INUM
      JFLOW(GROUP,IUNIT)=INUM
      CALL NUMUNIT(ITYPE(NUMBER,3),ITYPE(NUMBER,4))
      GOTO 33
   36 PRINT*,'     '
      PRINT*,'ENTER QUALITY OF WATER RECOVERED'
      PRINT*,' 1. POTABLE WATER'
      PRINT*,' 2. HYGIENE WATER'
      READ(5,*,ERR=36) IQUAL
      IF(IQUAL.LT.1.OR.IQUAL.GT.2) GOTO 36
      NUMB=NUMB+1
      ITYPE(NUMB,2)=IQUAL
      GOTO 35
   37 PRINT*,'    '
      PRINT*,'    '
      PRINT*,'DO YOU WISH TO SPECIFY ANY SECONDARY FLOW PATHS? (Y/N)'
   44 READ(5,11,ERR=37) ANSWER
   11 FORMAT(1A)
      IF (ANSWER.EQ.'N') GOTO 50
      IGROUP=GROUP
      IIUNIT=IUNIT
      I=0
   28 I=I+1
        J=0
   29   J=J+1
          IF (JFLOW(I,J).EQ.31) SSNAME='VCD'
          IF (JFLOW(I,J).EQ.32) SSNAME='TIMES'
          IF (JFLOW(I,J).EQ.33) SSNAME='AIR EVAP'
          IF (JFLOW(I,J).EQ.34) SSNAME='VPCAR'
          IF (JFLOW(I,J).EQ.41) SSNAME='MULTIFILTRATION'
          IF (JFLOW(I,J).EQ.42) SSNAME='HYPERFILTRATION'
          IF (JFLOW(I,J).EQ.43) SSNAME='REVERSE OSMOSIS'
          IL=(I*10)+J
      PRINT*,'    '
      PRINT*,'THE FIRST DIGIT OF THE NUMBER IN PARENTHESES REPRESENTS'
      PRINT*,'THE FLOW GROUP NUMBER AND THE SECOND DIGIT REPRESENTS THE'
      PRINT*,'NUMBER OF THE SUBSYSTEM IN THAT FLOW PATH.'
      PRINT*,'    '
      PRINT*,'DO YOU WISH TO PROCESS THE SECONDARY FLOW COMING OUT OF'
      PRINT 57,IL,SSNAME
   57 FORMAT(1X,'THE (',I2,') ',A15,' SUBSYSTEM? (Y/N)')
          ANSWER='N'
   47     READ(5,11,ERR=47) ANSWER
          IF (ANSWER.EQ.'N') GOTO 46
          PRINT*,'    '
          PRINT*,'WILL THIS FLOW STREAM BE GOING INTO A PREDEFINED ',
     *           'SUBSYSTEM? (Y/N)'
          ANSWER='N'
   48     READ(5,11,ERR=48) ANSWER
          IF (ANSWER.EQ.'Y') THEN
            PRINT*,'    '
   62       PRINT*,'WHICH OF THE FOLLOWING SUBSYSTEMS WILL THE FLOW ',
     *             'BE ENTERING?'
          PRINT*,'    '
          PRINT*,'THE FIRST DIGIT REPRESENTS THE FLOW GROUP NUMBER ',
     *           'AND THE SECOND DIGIT REPRESENTS THE NUMBER OF THE ',
     *           'SUBSYSTEM IN THAT FLOW PATH.'
          PRINT*,'    '
          DO 49 K=1,GROUP
            DO 49 L=1,IUNIT
```

```
            KL=(K*10)+L
            IF(JFLOW(K,L).EQ.31) PRINT*,' ',KL,'.  VCD'
            IF(JFLOW(K,L).EQ.32) PRINT*,' ',KL,'.  TIMES'
            IF(JFLOW(K,L).EQ.33) PRINT*,' ',KL,'.  AIR EVAP'
            IF(JFLOW(K,L).EQ.34) PRINT*,' ',KL,'.  VPCAR'
            IF(JFLOW(K,L).EQ.41) PRINT*,' ',KL,'.  MULTIFILTRATION'
            IF(JFLOW(K,L).EQ.42) PRINT*,' ',KL,'.  HYPERFILTRATION'
            IF(JFLOW(K,L).EQ.43) PRINT*,' ',KL,'.  REVERSE OSMOSIS'
49        CONTINUE
43        READ(5,*,ERR=43) INUM3
          KFLOW(I,J)=INUM3
          GOTO 46
          ENDIF
          PRINT*,'    '
          PRINT*,'    '
51        PRINT*,'WHAT UNIT WILL THIS FLOW STREAM BE ENTERING?'
          PRINT*,' 31.  VCD'
          PRINT*,' 32.  TIMES'
          PRINT*,' 33.  AIR EVAP'
          PRINT*,' 34.  VPCAR'
          PRINT*,' 41.  MULTIFILTRATION'
          PRINT*,' 42.  HYPERFILTRATION'
          PRINT*,' 43.  REVERSE OSMOSIS'
          READ(5,*,ERR=51) JNUM
          IF(JNUM.LT.31.OR.JNUM.GT.43) GOTO 51
          IF(JNUM.GT.34.AND.JNUM.LT.41) GOTO 51
          IF(JNUM.EQ.34) THEN
            PRINT*,'SUBSYSTEM IS NOT CURRENTLY MODELLED'
            GOTO 51
          ENDIF
          LFLOW(I,J)=JNUM
          NUMBER=NUMBER+1
          IUNIT=IUNIT+1
          IF(ANSWER.EQ.'P') GOTO 60
          GROUP=GROUP+1
60        ITYPE(NUMBER,1)=JNUM
          JFLOW(GROUP,IUNIT)=JNUM
          CALL NUMUNIT(ITYPE(NUMBER,3),ITYPE(NUMBER,4))
          PRINT*,'    '
52        PRINT*,'WILL THIS WATER BE PROCESSED AGAIN OR STORED? (P/S)'
          READ(5,11,ERR=52) ANSWER
          IF(ANSWER.EQ.'S') THEN
            TANKED='STORED'
53          PRINT*,'    '
            PRINT*,'ENTER QUALITY OF WATER RECOVERED'
            PRINT*,'   1. POTABLE WATER'
            PRINT*,'   2. HYGIENE WATER'
            READ(5,*,ERR=53) IQUAL
            IF (IQUAL.LT.1.OR.IQUAL.GT.2) GOTO 53
            NUMB=NUMB+1
            ITYPE(NUMB,2)=IQUAL
          ENDIF
          IF(ANSWER.EQ.'P') THEN
61          PRINT*,'WILL THE PRIMARY FLOW FROM THIS NEW UNIT BE ',
     *           'ENTERING A PREDEFINED SUBSYSTEM? (Y/N)'
            READ(5,11,ERR=61) ANSWER
            IF(ANSWER.EQ.'Y') THEN
              ANSWER='P'
              GOTO 62
            ELSE
```

```
              ANSWER='2'
              J=J+1
              GOTO 51
            ENDIF
          ENDIF
   46 II=J+1
      IF (JFLOW(I,II).NE.0) GOTO 29
      JJ=I+1
      IF (JFLOW(JJ,1).NE.0) GOTO 28
      IF (JFLOW(JJ,II).NE.0) THEN
        I=I+1
        GOTO 29
      ENDIF
   50 CALL AUXLSTRT
   55 PRINT*,'     '
      PRINT*,'     '
      PRINT*,'ENTER 1 TO SEE/CHANGE AVERAGE DESIGN LOADS'
      PRINT*,'ENTER 2 TO SEE/CHANGE SUBSYSTEM DATA'
      PRINT*,'ENTER 3 TO SEE/CHANGE AUXILIARY EQUIPMENT DATA'
      PRINT*,'ENTER 4 TO SEE SYSTEM ASSESSMENT RESULTS'
      PRINT*,'ENTER 5 TO SELECT NEW SUBSYSTEMS, USING DATABASE ',
     * 'DESIGN LOADS & SUBSYSTEM DATA'
      PRINT*,'ENTER 6 TO SELECT NEW SUBSYSTEMS, KEEPING CURRENT ',
     * 'DESIGN LOADS & SUBSYSTEM DATA'
      PRINT*,'ENTER 7 TO COMPARE SYSTEMS'
      PRINT*,'ENTER 8 TO QUIT'
      READ(5,*,ERR=55) IDECID
      IF (IDECID.LT.1.OR.IDECID.GT.8) GOTO 55
      IF (IDECID.EQ.1) CALL SBRSGN(6)
      IF (IDECID.EQ.2) CALL SBRSSDAT(6)
      IF (IDECID.EQ.3) CALL SBRAUXIL(6)
      IF (IDECID.EQ.4) CALL ANALIZE
      IF (IDECID.EQ.5) GOTO 5
      IF (IDECID.EQ.6) GOTO 4
      IF (IDECID.EQ.7) CALL COMPARE
      IF (IDECID.EQ.8) GOTO 9999
      GOTO 55
 9999 CLOSE(UNIT=1)
      STOP
      END
C
      SUBROUTINE ANALIZE
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
```

```
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
     * CLOTHW,
     * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMAT(3)
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),
     * COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      COMMON /FILE/ FNAME(NSYS)
      COMMON /IWRT/ IWRITE
      CHARACTER*12 FNAME
      CHARACTER*6 ADUMP
      DIMENSION COUNT(2)
C
C  SUBSYS(I,1-9) = SUBSYSTEM DATA FOR EACH SUBSYSTEM CHOSEN
C
C  SUBSYS( 0,J)  = LOAD (LB/DAY)
C  SUBSYS( 1,J)  = AVERAGE POWER, LIGHT SIDE (WATTS)
C  SUBSYS( 2,J)  = AVERAGE POWER, DARK SIDE (WATTS)
C  SUBSYS( 3,J)  = HEAT REJECTION (WATTS)
C  SUBSYS( 4,J)  = FLIGHT UNIT WEIGHT (LB)
C  SUBSYS( 5,J)  = FLIGHT UNIT VOLUME (FT3)
C  SUBSYS( 6,J)  = WEIGHT OF SPARES (LB)
C  SUBSYS( 7,J)  = VOLUME OF SPARES (LB)
C  SUBSYS( 8,J)  = WEIGHT OF CONSUMABLES (LB)
C  SUBSYS( 9,J)  = VOLUME OF CONSUMABLES (FT3)
C  SUBSYS(10,J)  = RELIABILITY (1-8)
C  SUBSYS(11,J)  = TECHNOLOGY READINESS (1-8)
C  SUBSYS(12,J)  = PACING TECHNOLOGY PROBLEMS (1-8)
C  SUBSYS(13,J)  = SAFETY (1-8)
C  SUBSYS(14,J)  = MAINTAINABILITY (1-8)
C  SUBSYS(15,J)  = COMMONALITY (1-8)
C  SUBSYS(16,J)  = MAINTENANCE TIME (HR)
C  SUBSYS(17,J)  = AVERAGE POWER OVER ORBIT (WATTS)
C  SUBSYS(18,J)  = AUXILIARY EQUIPMENT WEIGHT(LB)- EXCL. LOST MATL & STORAGE
C  SUBSYS(19,J)  = AUXILIARY EQUIPMENT VOLUME(FT3)-EXCL. LOST MATL & STORAGE
C  SUBSYS(20,J)  = AUXILIARY EQUIPMENT POWER (WATTS)
C  SUBSYS(21,J)  = AUX EQUIP-(LOST) MATERIAL WEIGHT (LB)
C  SUBSYS(22,J)  = AUX EQUIP-(LOST) MATERIAL STORAGE EQUIPMENT (LB)
C  SUBSYS(23,J)  = AUX EQUIP-(LOST) MATERIAL STORAGE EQUIPMENT (FT3)
C  SUBSYS(24,J)  = REDUNDANT UNIT WEIGHT(LB)
C  SUBSYS(25,J)  = REDUNDANT UNIT VOLUME(FT3)
C  SUBSYS(26,J)  = WATER WEIGHT, INITIAL STARTUP (LB)
C  SUBSYS(27,J)  = OVERALL RATING (1-8)
C  SUBSYS(28,J)  = INITIAL LAUNCH WEIGHT (LB)
C  SUBSYS(29,J)  = ON ORBIT VOLUME (LB)
C  SUBSYS(30,J)  = RESUPPLY WEIGHT (LB)
C  SUBSYS(31,J)  = RESUPPLY VOLUME (FT3)
C  SUBSYS(32,J)  = EQUIVALENT WEIGHT (LB)
C  SUBSYS(33,J)  = LIFETIME WEIGHT (LB)
C  SUBSYS(34,J)  = LIFETIME VOLUME (FT3)
C  SUBSYS(35,J)  = TOTAL POWER REQUIRED(WATTS)
C
C  TOTALS(1-35,J)= SAME AS SUBSYS ARRAY
```

```
C
C  TOTALS(I, 1)  = ARS TOTALS
C  TOTALS(I, 2)  = WRS TOTALS
C  TOTALS(I, 3)  = OVERALL TOTALS
C
      N=DESIGN(1)
      R=DESIGN(2)
      RE=DESIGN(10)
      M=DESIGN(3)
      CALL ARS
      CALL WRS
      CALL RESUP
      ALIGHT= DESIGN(55)/(DESIGN(55)+DESIGN(53))
      ICNTW=3
      DO 2 J=1,NUMBER
      ISYS=ITYPE(J,1)
      IF (ISYS.GE.21.AND.ISYS.LE.30) SUBSYS(0,J)=O2TOT
      IF (ISYS.GE. 1.AND.ISYS.LE.10) SUBSYS(0,J)=CO2MET
      IF (ISYS.GE.11.AND.ISYS.LE.20) SUBSYS(0,J)=CO2TOT
      SUBSYS(17,J)=SUBSYS(1,J)*ALIGHT+SUBSYS(2,J)*(1.-ALIGHT)
      IF (ISYS.GE.21.AND.ISYS.LE.30) IAUX=1
      IF (ISYS.GE. 1.AND.ISYS.LE.10) IAUX=2
      IF (ISYS.GE.11.AND.ISYS.LE.20) IAUX=3
      IF (ISYS.GE.31) THEN
        ICNTW=ICNTW+1
        IAUX=ICNTW
      ENDIF
      SUBSYS(18,J)=AUXEQ(IAUX,1,1)+AUXEQ(IAUX,2,1)
      SUBSYS(19,J)=AUXEQ(IAUX,1,2)+AUXEQ(IAUX,2,2)
      SUBSYS(20,J)=AUXEQ(IAUX,1,3)+AUXEQ(IAUX,2,3)
      IF (ISYS.GE.11.AND.ISYS.LE.20) THEN
        SUBSYS(21,J)=AUXMAT(1)
      ELSE
        SUBSYS(21,J)=AUXEQ(IAUX,3,1)+AUXEQ(IAUX,4,1)
      ENDIF
      SUBSYS(22,J)=AUXEQ(IAUX,5,1)+AUXEQ(IAUX,6,1)
      SUBSYS(23,J)=AUXEQ(IAUX,5,2)+AUXEQ(IAUX,6,2)
      SUBSYS(24,J)=SUBSYS(4,J)*ITYPE(J,4)/ITYPE(J,3)
      SUBSYS(25,J)=SUBSYS(5,J)*ITYPE(J,4)/ITYPE(J,3)
      IF (ISYS.GE.31) SUBSYS(26,J)=AUXEQ(IAUX,0,1)
      SUBSYS(27,J)=(SUBSYS(10,J)*SUBSYS(11,J)*SUBSYS(12,J)*SUBSYS(13,J)*
     *      SUBSYS(14,J)*SUBSYS(15,J))**(1./6.)
      SUBSYS(28,J)=SUBSYS(4,J)+SUBSYS(6,J)+SUBSYS(8,J)+(SUBSYS(18,J)+
     *      SUBSYS(22,J))+SUBSYS(24,J)+SUBSYS(26,J)
      SUBSYS(29,J)=SUBSYS(5,J)+SUBSYS(7,J)+SUBSYS(9,J)+SUBSYS(19,J)+
     *      SUBSYS(23,J)+SUBSYS(25,J)
      SUBSYS(30,J)=SUBSYS(6,J)+SUBSYS(8,J)+SUBSYS(21,J)+SUBSYS(22,J)
      SUBSYS(31,J)=SUBSYS(7,J)+SUBSYS(9,J)+SUBSYS(23,J)
      SUBSYS(32,J)=SUBSYS(28,J)+SUBSYS(21,J)+(SUBSYS(1,J)*ALIGHT*
     * DESIGN(52)+SUBSYS(2,J)*(1.-ALIGHT)*DESIGN(53))+SUBSYS(3,J)*
     * DESIGN(54)+SUBSYS(20,J)*(ALIGHT*DESIGN(52)+(1.-ALIGHT)*
     * DESIGN(53))
      SUBSYS(33,J)=SUBSYS(4,J)+SUBSYS(18,J)+
     * SUBSYS(24,J)+SUBSYS(26,J)+SUBSYS(30,J)*ANINT(M*365./R)
      SUBSYS(34,J)=SUBSYS(5,J)+SUBSYS(19,J)+SUBSYS(25,J)+
     * SUBSYS(31,J)*ANINT(M*365./R)
      SUBSYS(35,J)=SUBSYS(17,J)+SUBSYS(20,J)
    2 CONTINUE
      DO 6 J=1,3
      DO 5 I=1,NSUBSYS
```

```
      TOTALS(I,J)=0.
    5 CONTINUE
    6 CONTINUE
      COUNT(1)=0.
      COUNT(2)=0.
      DO 20 I=1,NUMBER
      IF (ITYPE(I,1).LE.30) K=1
      IF (ITYPE(I,1).GE.31) K=2
      COUNT(K)=COUNT(K)+1.
      DO 10 J=1,NSUBSYS
      IF (J.LE.9.OR.(J.GE.16.AND.J.LE.17).OR.(J.GE.24.AND.J.LE.25))
     +    TOTALS(J,K)=TOTALS(J,K)+SUBSYS(J,I)
      IF ((J.GE.10.AND.J.LE.15).OR.J.EQ.27) THEN
        IF (TOTALS(J,K).EQ.0.) TOTALS(J,K)=1.
        TOTALS(J,K)=TOTALS(J,K)*SUBSYS(J,I)
      ENDIF
   10 CONTINUE
   20 CONTINUE
      DO 30 K=1,2
      DO 25 J=10,15
      IF (COUNT(K).EQ.0.) GOTO 25
      IF (TOTALS(J,K).NE.0.) TOTALS(J,K)=TOTALS(J,K)**(1./COUNT(K))
   25 CONTINUE
      IF (COUNT(K).EQ.0.) GOTO 30
      TOTALS(27,K)=TOTALS(27,K)**(1./COUNT(K))
   30 CONTINUE
      DO 40 J=1,NSUBSYS
      IF (J.LE.9.OR.(J.GE.16.AND.J.LE.17).OR.(J.GE.24.AND.J.LE.25))
     *    TOTALS(J,3)=TOTALS(J,1)+TOTALS(J,2)
      IF ((J.GE.10.AND.J.LE.15).OR.J.EQ.27) TOTALS(J,3)=(TOTALS(J,1)*
     *       TOTALS(J,2))**.5
   40 CONTINUE
      TOTALS(18,1)=AUXEQ(16,1,1)+AUXEQ(16,2,1)
      TOTALS(18,2)=AUXEQ(17,1,1)+AUXEQ(17,2,1)
      TOTALS(18,3)=TOTALS(18,1)+TOTALS(18,2)
      TOTALS(19,1)=AUXEQ(16,1,2)+AUXEQ(16,2,2)
      TOTALS(19,2)=AUXEQ(17,1,2)+AUXEQ(17,2,2)
      TOTALS(19,3)=TOTALS(19,1)+TOTALS(19,2)
      TOTALS(20,1)=AUXEQ(16,1,3)+AUXEQ(16,2,3)+
     *  AUXEQ(16,5,3)+AUXEQ(16,6,3)
      TOTALS(20,2)=AUXEQ(17,1,3)+AUXEQ(17,2,3)+
     *  AUXEQ(17,5,3)+AUXEQ(17,6,3)
      TOTALS(20,3)=TOTALS(20,1)+TOTALS(20,2)
      TOTALS(21,1)=AUXEQ(16,3,1)+AUXEQ(16,4,1)
      TOTALS(21,2)=AUXEQ(17,3,1)+AUXEQ(17,4,1)
      TOTALS(21,3)=TOTALS(21,1)+TOTALS(21,2)
      TOTALS(22,1)=AUXEQ(16,5,1)+AUXEQ(16,6,1)
      TOTALS(22,2)=AUXEQ(17,5,1)+AUXEQ(17,6,1)
      TOTALS(22,3)=TOTALS(22,1)+TOTALS(22,2)
      TOTALS(23,1)=AUXEQ(16,5,2)+AUXEQ(16,6,2)
      TOTALS(23,2)=AUXEQ(17,5,2)+AUXEQ(17,6,2)
      TOTALS(23,3)=TOTALS(23,1)+TOTALS(23,2)
      TOTALS(26,1)=AUXEQ(16,0,1)
      TOTALS(26,2)=AUXEQ(17,0,1)
      TOTALS(26,3)=TOTALS(26,1)+TOTALS(26,2)
      DO 50 K=1,3
      TOTALS(28,K)=TOTALS(4,K,+TOTALS(6,K)+TOTALS(8,K)+TOTALS(18,K)+
     * TOTALS(22,K)+TOTALS(24,K)+TOTALS(26,K)
      TOTALS(29,K)=TOTALS(5,K)+TOTALS(7,K)+TOTALS(9,K)+TOTALS(19,K)+
     * TOTALS(23,K)+TOTALS(25,K)
```

```
      TOTALS(30,K)=TOTALS(6,K)+TOTALS(8,K)+TOTALS(21,K)+TOTALS(22,K)
      TOTALS(31,K)=TOTALS(7,K)+TOTALS(9,K)+TOTALS(23,K)
      TOTALS(32,K)=TOTALS(28,K)+TOTALS(21,K)+(TOTALS(1,K)*ALIGHT*
     * DESIGN(52)+TOTALS(2,K)*(1.-ALIGHT)*DESIGN(53))+TOTALS(3,J)*
     * DESIGN(54)+TOTALS(20,J)*(ALIGHT*DESIGN(52)+(1.-ALIGHT)*
     * DESIGN(53))
      TOTALS(33,K)=TOTALS(4,K)+TOTALS(18,K)+TOTALS(24,K)+
     * TOTALS(26,K)+TOTALS(30,K)*ANINT(M*365./R)
      TOTALS(34,K)=TOTALS(5,K)+TOTALS(19,K)+TOTALS(25,K)+
     * TOTALS(31,K)*ANINT(M*365./R)
      TOTALS(35,K)=TOTALS(17,K)+TOTALS(20,K)
   50 CONTINUE
      IF (IWRITE.EQ.2) GOTO 45
      NWRITE=1
      WRITE(NWRITE,41)
   41 FORMAT(80('*'),/,80('*'))
      CALL SRBSUB(NWRITE)
      CALL WARSMAT(NWRITE)
      CALL WWRSMAT(NWRITE)
      CALL WRESUP(NWRITE)
      CALL SBRAUXIL(NWRITE)
      CALL SBRSGN(NWRITE)
      CALL SBRSSDAT(NWRITE)
      WRITE(NWRITE,42)
   42 FORMAT(4(/))
   45 IF (IWRITE.EQ.1) GOTO 90
      NWRITE=6
      CALL SRBSUB(NWRITE)
      CALL WARSMAT(NWRITE)
      CALL WWRSMAT(NWRITE)
      CALL WRESUP(NWRITE)
   90 CALL STORE
      RETURN
      END
C
      SUBROUTINE ARS
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
```

```
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
     * CLOTHW,
     * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMAT(3)
      CHARACTER*6 ADUMP
C
      PPN2=DESIGN(4)
      PPO2=DESIGN(5)
      PTOT=PPO2+PPN2
      N2LEAK=(PPN2/PTOT)*DESIGN(6)
      N2AIR=(PPN2/PTOT)*DESIGN(7)
      N2RPRS=0.
      N2COM=0.
      N2PURG=0.
      N2TOT=N2LEAK+N2AIR+N2RPRS+N2COM+N2PURG
C
      CO2MET=DESIGN(9)*N
      CO2EVA=DESIGN(12)*RE/R
      CO2TOT=CO2MET+CO2EVA
C
C  O2 GEN REQUIREMENTS (SIZING BASED ON O2 GEN)
      E1=-1.0
      CALL SBRO2GEN(PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,O2COM,
     *              O2EDC,O2TOT,CO2MET)
      DO 10 I=1,NUMBER
      IF (ITYPE(I,1).EQ.21) CALL SFWES(I,E1,O2TOT)
      IF (ITYPE(I,1).EQ.22) CALL SPOLLF(I,E1,O2TOT)
      IF (ITYPE(I,1).EQ.23) CALL SPOLSF(I,E1,O2TOT)
      IF (ITYPE(I,1).EQ.24) CALL O2STOR(I,E1,O2TOT)
   10 CONTINUE
      H2TOT=2./16.*O2TOT
      H2OGEN=18./16.*O2TOT
      IF (E1.EQ.-1.0) H2TOT=0.
      IF (E1.EQ.-1.0) H2OGEN=0.
C
C  CO2 REMOVAL REQUIREMENTS (SIZING BASED ON CO2 PRODUCED)
      H2EDC=0.
      H2OREM=0.
      H2OREMRQ=0.
      DO 20 I=1,NUMBER
      IF (ITYPE(I,1).EQ.1) CALL EDC(I,CO2MET,O2EDC,H2EDC,H2OREM)
      IF (ITYPE(I,1).EQ.2) CALL SAWD(I,CO2MET,H2OREM,H2OREMRQ)
      IF (ITYPE(I,1).EQ.3) CALL HMOLSV(I,CO2MET,H2OREM)
      IF (ITYPE(I,1).EQ.4) CALL LIOH(I,CO2MET,H2OREM)
      IF (ITYPE(I,1).EQ.5) CALL FBDMSV(I,CO2MET,H2OREM)
      IF (ITYPE(I,1).EQ.6) CALL SKLMSV(I,CO2MET,H2OREM)
   20 CONTINUE
C
C  CO2 REDUCTION REQUIREMENTS (SIZING BASED ON CO2 PRODUCED)
      H2AVAL=H2TOT-H2EDC
      H2BSCH=0.
      H2SAB=0.
      C=0.
      CH4=0.
      CO2RED=0.
      H2ORED=0.
      DO 30 I=1,NUMBER
      IF (ITYPE(I,1).EQ.11) CALL BOSCH(I,CO2TOT,H2AVAL,H2ORED,CO2RED,
     *                                 C,H2BSCH)
```

```
      IF (ITYPE(I,1).EQ.12) CALL CABAT(I,CO2TOT,H2AVAL,H2ORED,CO2RED,
     *                                 CH4,H2SAB)
      IF (ITYPE(I,1).EQ.13) CALL SABCH4(I,CO2TOT,H2AVAL,H2ORED,CO2RED,
     *                                 C,H2SAB)
   30 CONTINUE
      CO2LST=CO2TOT-CO2RED
      H2RED=AMAX1(H2BSCH,H2SAB)
      H2REQ=H2RED+H2EDC
      H2STOR=AMAX1(0.,H2AVAL-H2RED)
      H2STORR=H2STOR*R
      CRES=C*R
      CH4R=CH4*R
      CO2LSTR=CO2LST*R
      H2OARS=H2OGEN+H2OREMRQ-H2OREM-H2ORED
      RETURN
      END
C
      SUBROUTINE WRS
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP,TANKED
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
     * CLOTHW,
     * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMAT(3),
     * EFF(NWAUX),SSWATER(NWAUX),TSWATER(NWAUX)
       DIMENSION WATERS(NWAUX),TWATER(NWAUX),EF(NWAUX,NWAUX)
      CHARACTER*6 ADUMP
      CHARACTER*6 TANKED
C
      HAND=DESIGN(28)*N
      SHOWER=DESIGN(29)*N
      CLOTHES=DESIGN(30)*N
      DISHW=DESIGN(31)*N
      DRINK=DESIGN(15)*N
      FOODP=DESIGN(16)*N
      URINFL=DESIGN(26)*N
      EXPH2O=DESIGN(32)*N
      EVAH2O=DESIGN(13)*N
```

```
      POTREQ=DRINK+FOODP
      H2OREQ=HAND+SHOWER+CLOTHES+DISHW+DRINK+FOODP+URINFL+EXPH2O
     *       +EVAH2O+H2OREMRQ+H2OGEN
      H2OMET=DESIGN(18)*N
      H2OFOOD=DESIGN(17)*N
      H2OPROD=H2OMET+H2OFOOD
      H2OTOT=H2OREQ+H2OPROD
      SWEAT=DESIGN(24)*N
      HYGIENL=DESIGN(19)*N
      FOODPL=DESIGN(20)*N
      LAUNL=DESIGN(22)*N
      DISHWL=DESIGN(23)*N
      EXPH2OL=DESIGN(21)*N
      COND=SWEAT+HYGIENL+FOODPL+LAUNL+DISHWL+H2OREM+EXPH2OL
      FECAL=DESIGN(27)*N
C EVA+EXP H2O LOST
      EVAH2OW=DESIGN(14)*N + DESIGN(33)*N
C EVA+EXP H2O LOAD
      EVAH2OLD=(EVAH2O+EXPH2O)-EVAH2OW-EXPH2OL
      REDH2O=H2ORED
      HANDLD=HAND-HYGIENL*(HAND/(HAND+SHOWER))
      SHOWLD=SHOWER-HYGIENL*(SHOWER/(HAND+SHOWER))
      HYGIENLD=HANDLD+SHOWLD
      CLOTHLD=CLOTHES-LAUNL
      H2OURN=DESIGN(25)*N
      URINLD=URINFL+H2OURN+EVAH2OLD
      DISHWLD=DISHW-DISHWL
C
      DO 5 I=1,NWAUX
        DO 5 J=1,NWAUX
    5     E(I,J)=0.
      DO 6 I=1,NWAUX
        EFF(I)=0.
        SSWATER(I)=0.
        WATERS(I)=0.
        TWATER(I)=0.
        TSWATER(I)=0.
    6   WATER(I)=0.
      DO 7 I=1,NUMBER
        IF(ITYPE(I,1).EQ.31) EFF(I)=0.96
        IF(ITYPE(I,1).EQ.32) EFF(I)=0.939
        IF(ITYPE(I,1).EQ.33) EFF(I)=1.0
        IF(ITYPE(I,1).EQ.34) EFF(I)=0.0
        IF(ITYPE(I,1).EQ.41) EFF(I)=1.0
        IF(ITYPE(I,1).EQ.42) EFF(I)=0.92
        IF(ITYPE(I,1).EQ.43) EFF(I)=0.96
    7 CONTINUE
      DO 10 I=1,GROUP
      IF (IWATER(1).EQ.I) WATER(I)=WATER(I)+COND
      IF (IWATER(2).EQ.I) WATER(I)=WATER(I)+REDH2O
      IF (IWATER(3).EQ.I) WATER(I)=WATER(I)+HYGIENLD
      IF (IWATER(4).EQ.I) WATER(I)=WATER(I)+CLOTHLD
      IF (IWATER(5).EQ.I) WATER(I)=WATER(I)+URINLD
      IF (IWATER(6).EQ.I) WATER(I)=WATER(I)+DISHWLD
      TWATER(I)=WATER(I)
   10 CONTINUE
C
      CONDR=0.
      REDH2OR=0.
      HYGIENR=0.
```

```
      CLOTHR=0.
      URINR=0.
      DISHWR=0.
      ZCONDR=COND
      ZREDH2OR=REDH2O
      ZHYGIENR=HYGIENLD
      ZCLOTHR=CLOTHLD
      ZURINR=URINLD
      ZDISHWR=DISHWLD
C
      DO 20 I=1,GROUP
        DO 30 J=1,IUNIT
          NUM=(NUMBER-IUNIT)+J
          IF(JFLOW(I,J).EQ.31) THEN
            CALL VCD(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.32) THEN
            CALL TIMES(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.33) THEN
            CALL AIREVP(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.34) THEN
            CALL VPCAR(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.41) THEN
            CALL MF(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.42) THEN
            CALL HF(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
          IF(JFLOW(I,J).EQ.43) THEN
            CALL RO(NUM,WATER(I),E(I,J))
            SUBSYS(0,NUM)=WATER(I)
            SWATER(I,J)=WATER(I)*(1.0-E(I,J))
            WATER(I)=WATER(I)*E(I,J)
          ENDIF
   30   CONTINUE
   20 CONTINUE
C
      CONDP=0.
      REDH2OP=0.
```

```
      HYGIENP=0.
      CLOTHP=0.
      URINP=0.
      DISHWP=0.
C
      DO 18 K=1,GROUP
        DO 19 L=1,NWNUM
          IF(IWATER(L).EQ.K) GOTO 18
   19   CONTINUE
          GOTO 15
   18 CONTINUE
   15   DO 17 M=1,IUNIT
          IF(JFLOW(K,M).NE.0) GOTO 16
   17   CONTINUE
   16   NUM=M+(NUMBER-IUNIT)
      DO 60 I=1,GROUP
        DO 60 J=1,IUNIT
          III=(NUMBER-IUNIT)+I
          IF(LFLOW(I,J).EQ.31) THEN
            CALL VCD(NUM,SWATER(I,J),EF(I,J))
            SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
            SWATER(I,J)=SWATER(I,J)*EF(I,J)
            IF(TANKED.EQ.'STORED') THEN
            IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
            IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
            ENDIF
            NUM=NUM+1
          ENDIF
          IF(LFLOW(I,J).EQ.32) THEN
            CALL TIMES(NUM,SWATER(I,J),EF(I,J))
            SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
            SWATER(I,J)=SWATER(I,J)*EF(I,J)
            IF(TANKED.EQ.'STORED') THEN
            IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
            IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
            ENDIF
            NUM=NUM+1
          ENDIF
          IF(LFLOW(I,J).EQ.33) THEN
            CALL AIREVP(NUM,SWATER(I,J),EF(I,J))
            SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
            SWATER(I,J)=SWATER(I,J)*EF(I,J)
            IF(TANKED.EQ.'STORED') THEN
            IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
            IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
            ENDIF
            NUM=NUM+1
          ENDIF
          IF(LFLOW(I,J).EQ.34) THEN
            CALL VPCAR(NUM,SWATER(I,J),EF(I,J))
            SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
            SWATER(I,J)=SWATER(I,J)*EF(I,J)
            IF(TANKED.EQ.'STORED') THEN
            IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
            IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
            ENDIF
            NUM=NUM+1
          ENDIF
          IF(LFLOW(I,J).EQ.41) THEN
            CALL MF(NUM,SWATER(I,J),EF(I,J))
```

```
          SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
          SWATER(I,J)=SWATER(I,J)*EF(I,J)
          IF(TANKED.EQ.'STORED') THEN
          IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
          IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
          ENDIF
          NUM=NUM+1
        ENDIF
        IF(LFLOW(I,J).EQ.42) THEN
          CALL HF(NUM,SWATER(I,J),EF(I,J))
          SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
          SWATER(I,J)=SWATER(I,J)*EF(I,J)
          IF(TANKED.EQ.'STORED') THEN
          IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
          IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
          ENDIF
          NUM=NUM+1
        ENDIF
        IF(LFLOW(I,J).EQ.43) THEN
          CALL RO(NUM,SWATER(I,J),EF(I,J))
          SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
          SWATER(I,J)=SWATER(I,J)*EF(I,J)
          IF(TANKED.EQ.'STORED') THEN
          IF(ITYPE(III,2).EQ.1) SSWATER(I)=SWATER(I,J)
          IF(ITYPE(III,2).EQ.2) TSWATER(I)=SWATER(I,J)
          ENDIF
          NUM=NUM+1
        ENDIF
60  CONTINUE
    DO 40 I=1,GROUP
      DO 40 J=1,IUNIT
        IF(KFLOW(I,J).NE.0) THEN
          III=KFLOW(I,J)/10
          JJJ=KFLOW(I,J)-(III*10)
          DO 50 K=JJJ,IUNIT
            NUM=(NUMBER-IUNIT)+K
            IF(JFLOW(III,K).EQ.0) GOTO 50
            SUBSYS(0,NUM)=SWATER(I,J)+SUBSYS(0,NUM)
            IF(E(III,K).NE.0) SWATER(I,J)=SWATER(I,J)*E(III,K)
50        CONTINUE
          PSWATER(III)=SWATER(I,J)
          WATERS(I)=SWATER(I,J)
        ENDIF
40  CONTINUE
    DO 21 I=1,GROUP
      JJJ=(NUMBER-IUNIT)+I
      IF(IWATER(1).EQ.I) THEN
        DO 22 J=1,IUNIT
          JJ=(NUMBER-IUNIT)+J
22        IF(JFLOW(I,J).NE.0) ZCONDR=ZCONDR*EFF(JJ)
        IF(ITYPE(JJJ,2).EQ.1) CONDP=ZCONDR
       ZCONDR=ZCONDR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          COND/TWATER(I)
       CONDR=ZCONDR
      ENDIF
      IF(IWATER(2).EQ.I) THEN
        DO 23 J=1,IUNIT
          JJ=(NUMBER-IUNIT)+J
23        IF(JFLOW(I,J).NE.0) ZREDH2OR=ZREDH2OR*EFF(JJ)
        IF(ITYPE(JJJ,2).EQ.1) REDH2OP=ZREDH2OR
```

```
      ZREDH2OR=ZREDH2OR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          REDH2O/TWATER(I)
      REDH2OR=ZREDH2OR
        ENDIF
        IF(IWATER(3).EQ.I) THEN
          DO 24 J=1,IUNIT
            JJ=(NUMBER-IUNIT)+J
   24       IF(JFLOW(I,J).NE.0) ZHYGIENR=ZHYGIENR*EFF(JJ)
          IF(ITYPE(JJJ,2).EQ.1) HYGIENP=ZHYGIENR
      ZHYGIENR=ZHYGIENR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          HYGIENLD/TWATER(I)
      HYGIENR=ZHYGIENR
        ENDIF
        IF(IWATER(4).EQ.I) THEN
          DO 25 J=1,IUNIT
            JJ=(NUMBER-IUNIT)+J
   25       IF(JFLOW(I,J).NE.0) ZCLOTHR=ZCLOTHR*EFF(JJ)
          IF(ITYPE(JJJ,2).EQ.1) CLOTHP=ZCLOTHR
      ZCLOTHR=ZCLOTHR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          CLOTHLD/TWATER(I)
      CLOTHR=ZCLOTHR
        ENDIF
        IF(IWATER(5).EQ.I) THEN
          DO 26 J=1,IUNIT
            JJ=(NUMBER-IUNIT)+J
   26       IF(JFLOW(I,J).NE.0) ZURINR=ZURINR*EFF(JJ)
          IF(ITYPE(JJJ,2).EQ.1) URINP=ZURINR
      ZURINR=ZURINR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          URINLD/TWATER(I)
      URINR=ZURINR
        ENDIF
        IF(IWATER(6).EQ.I) THEN
          DO 27 J=1,IUNIT
            JJ=(NUMBER-IUNIT)+J
   27       IF(JFLOW(I,J).NE.0) ZDISHWR=ZDISHWR*EFF(JJ)
          IF(ITYPE(JJJ,2).EQ.1) DISHWP=ZDISHWR
      ZDISHWR=ZDISHWR+(WATERS(I)+SSWATER(I)+TSWATER(I))*
     *          DISHWLD/TWATER(I)
      DISHWR=ZDISHWR
        ENDIF
   21 CONTINUE
      POTREC=0.
      H2OREC=0.
      DO 70 I=1,GROUP
        KKK=(NUMBER-IUNIT)+I
        IF(ITYPE(KKK,2).EQ.1) THEN
          POTREC=POTREC+WATER(I)+PSWATER(I)+SSWATER(I)
        ELSE
          HYGREC=HYGREC+WATER(I)+PSWATER(I)+TSWATER(I)
        ENDIF
   70 CONTINUE
      CONDS=COND-CONDR
      REDH2OS=REDH2O-REDH2OR
      HYGIENS=HYGIENLD-HYGIENR
      CLOTHS=CLOTHLD-CLOTHR
      URINS=URINLD-URINR
      DISHWS=DISHWLD-DISHWR
      OTHERLD=EVAH2OW
      OTHERR=0.
      OTHERP=0.
```

```
      OTHERS=OTHERLD
      H2OLOAD=COND+REDH2O+HYGIENLD+CLOTHLD+URINLD+DISHWLD+OTHERLD+FECAL
      H2OREC=CONDR+REDH2OR+HYGIENR+CLOTHR+URINR+DISHWR
      H2OSTOR=H2OLOAD-H2OREC
      H2ONET=H2OTOT-H2OARS
      GREYRQ=H2OREQ-POTREQ
      GREYRC=H2OREC-POTREC+AMAX1(0.,POTREC-POTREQ)
      MAKEUP=AMAX1(0.,POTREQ-POTREC)+AMAX1(0.,GREYRQ-GREYRC)
      EXCESS=AMAX1(0.,GREYRC-GREYRQ)
      H2OSTORR=H2OSTOR*R
      MAKEUPR=MAKEUP*R
      EXCESSR=EXCESS*R
      RETURN
      END
C
      SUBROUTINE SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ANUNIT=FLOAT(ITYPE(ILOC,3))
      AMLTFCW=1.+(RATIO-1.0)*FACTORW
      AMLTFCV=1.+(RATIO-1.0)*FACTORV
      AMLTFCP=1.+(RATIO-1.0)*FACTORP
C AVG POWER,LIGHT(WATTS)
      SUBSYS(1,ILOC)=SSDATA( 1,ISYS)*AMLTFCP*ANUNIT
C CAVG POWER,DARK(WATTS)
      SUBSYS(2,ILOC)=SSDATA( 2,ISYS)*AMLTFCP*ANUNIT
C HEAT REJ(WATTS)
      SUBSYS(3,ILOC)=SSDATA( 3,ISYS)*AMLTFCP*ANUNIT
C FLIGHT UNIT(LB)
      SUBSYS(4,ILOC)=SSDATA( 4,ISYS)*AMLTFCW*ANUNIT
C FLIGHT UNIT(FT3)
      SUBSYS(5,ILOC)=SSDATA( 5,ISYS)*AMLTFCV*ANUNIT
C SPARES(LB)
      SUBSYS(6,ILOC)=SSDATA( 6,ISYS)*AMLTFCW*ANUNIT
C SPARES(FT3)
      SUBSYS(7,ILOC)=SSDATA( 7,ISYS)*AMLTFCV*ANUNIT
C CONS(LB)
      SUBSYS(8,ILOC)=SSDATA( 8,ISYS)*AMLTFCW*ANUNIT
C CONS(FT3)
      SUBSYS(9,ILOC)=SSDATA( 9,ISYS)*AMLTFCV*ANUNIT
C RELIABILITY(1-8)
      SUBSYS(10,ILOC)=SSDATA(10,ISYS)
C TECH READ(1-8)
      SUBSYS(11,ILOC)=SSDATA(11,ISYS)
C PAC TECH PROB(1-8)
      SUBSYS(12,ILOC)=SSDATA(12,ISYS)
C SAFETY(1-8)
      SUBSYS(13,ILOC)=SSDATA(13,ISYS)
```

```
C MAINTAINABILITY(1-8)
      SUBSYS(14,ILOC)=SSDATA(14,ISYS)
C COMMONALITY(1-8)
      SUBSYS(15,ILOC)=SSDATA(15,ISYS)
C MAINTENANCE (HR)
      SUBSYS(16,ILOC)=SSDATA(16,ISYS)*AMLTFCW*ANUNIT
      RETURN
      END
C
      SUBROUTINE SBRO2GEN(PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     *                    O2COM,O2EDC,O2TOT,CO2MET)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      O2MET=DESIGN(8)*N
      O2EVA=DESIGN(11)*RE/R
      O2LEAK=(PPO2/PTOT)*DESIGN(6)
      O2AIR=(PPO2/PTOT)*DESIGN(7)
      O2RPRS=0.
      O2COM=0.
      CALL EDCO2(CO2MET,O2EDC)
      O2TOT=O2MET+O2EVA+O2LEAK+O2AIR+O2RPRS+O2COM+O2EDC
      RETURN
      END
C
      SUBROUTINE SFWES(ILOC,E,O2GEN)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=21
      O2REF=SSDATA(18,ISYS)
      RATIO=(O2GEN/O2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
```

```
      END
C
      SUBROUTINE SPOLLF(ILOC,E,O2GEN)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=22
      O2REF=SSDATA(18,ISYS)
      RATIO=(O2GEN/O2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE SPOLSF(ILOC,E,O2GEN)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=23
      O2REF=SSDATA(18,ISYS)
      RATIO=(O2GEN/O2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE O2STOR(ILOC,E,O2GEN)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER(NCOMPAR=20)
      COMMON /CHAR/ ADUMP
```

```
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=24
      O2REF=SSDATA(18,ISYS)
      RATIO=(O2GEN/O2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=-1.0
      RETURN
      END
C
      SUBROUTINE EDC(ILOC,CO2MET,O2EDC,H2EDC,H2OREM)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=1
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2EDC=(SSDATA(21,ISYS)/R)*RATIO
      IF (H2EDC.EQ.0.) H2EDC=CO2MET*(2./44.)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      IF (H2OREM.EQ.0.) H2OREM=CO2MET*(18./44.)
      RETURN
      ENTRY EDCO2(CO2MET,O2EDC)
      ISYS=1
      DO 10 I=1,NUMBER
   10 IF (ITYPE(I,1).EQ.ISYS) GOTO 20
      O2EDC=0.
      GOTO 99
   20 CO2REF=SSDATA(19,ISYS)
      O2EDC=(SSDATA(22,ISYS)/R)*CO2MET/CO2REF
      IF (O2EDC.EQ.0.) O2EDC=CO2MET*(32./(44.*2.))
   99 RETURN
      END
C
      SUBROUTINE SAWD(ILOC,CO2MET,H2OREM,H2OREMRQ)
```

```
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=2
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      H2OREMRQ=H2OREM
      RETURN
      END
C
      SUBROUTINE HMOLSV(ILOC,CO2MET,H2OREM)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=3
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      RETURN
      END
C
      SUBROUTINE LIOH(ILOC,CO2MET,H2OREM)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
```

```
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=4
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      RETURN
      END
C
      SUBROUTINE FBDMSV(ILOC,CO2MET,H2OREM)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=5
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      RETURN
      END
C
      SUBROUTINE SKLMSV(ILOC,CO2MET,H2OREM)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=6
```

```
      CO2REF=SSDATA(19,ISYS)
      RATIO=(CO2MET/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2OREM=(SSDATA(23,ISYS)/R)*RATIO
      RETURN
      END
C
      SUBROUTINE BOSCH(ILOC,CO2TOT,H2AVAL,H2ORED,CO2RED,C,H2BSCH)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=11
      CO2REF=SSDATA(19,ISYS)
      H2REQ=(SSDATA(21,ISYS)/R)*CO2TOT/CO2REF
      IF (H2REQ.EQ.0.) H2REQ=CO2TOT*(2.*2./44.)
      IF (H2REQ.LE.H2AVAL) H2BSCH=H2REQ
      IF (H2REQ.GT.H2AVAL) H2BSCH=H2AVAL
      CO2RED=CO2TOT*H2BSCH/H2REQ
      RATIO=(CO2RED/CO2REF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2ORED=SSDATA(23,ISYS)/R*RATIO
      IF (H2ORED.EQ.0.) H2ORED=CO2RED*(2.*18./44.)
      C=SSDATA(24,ISYS)/R*RATIO
      IF (C.EQ.0.) C=CO2RED*(12./44.)
      RETURN
      END
C
      SUBROUTINE SABAT(ILOC,CO2TOT,H2AVAL,H2ORED,CO2RED,CH4,H2SAB)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=12
```

```
      CO2REF=SSDATA(19,ISYS)
      CO2RFN=SSDATA(26,ISYS)/R
      CO2RFR=CO2REF-CO2RFN
      H2REQ=(SSDATA(21,ISYS)/R)*CO2TOT/CO2RFR
      IF (H2REQ.EQ.0.) H2REQ=CO2TOT*(4.*2./44.)
      IF (H2REQ.LE.H2AVAL) H2SAB=H2REQ
      IF (H2REQ.GT.H2AVAL) H2SAB=H2AVAL
      CO2RED=CO2TOT*H2SAB/H2REQ
      RATIO=(CO2RED/CO2RFR)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      H2ORED=SSDATA(23,ISYS)/R*RATIO
      IF (H2ORED.EQ.0.) H2ORED=CO2RED*(2.*18./44.)
      CH4=SSDATA(25,ISYS)/R*RATIO
      IF (CH4.EQ.0.) CH4=CO2RED*(16./44.)
      RETURN
      END
C
      SUBROUTINE SABCH4(ILOC,CO2TOT,H2AVAL,H2ORED,CO2RED,C,H2SAB)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=13
      CO2REF=SSDATA(19,ISYS)
      RETURN
      END
C
      SUBROUTINE VCD(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=31
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
```

```
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE TIMES(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=32
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE AIREVP(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=33
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE VPCAR(ILOC,WATR,E)
      REAL N,M
```

```
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=34
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(20,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE MF(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=41
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE HF(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
```

```
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=42
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE RO(ILOC,WATR,E)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
      ISYS=43
      H2OREF=SSDATA(20,ISYS)
      RATIO=(WATR/H2OREF)/ITYPE(ILOC,3)
      FACTORW=SSDATA(27,ISYS)
      FACTORV=SSDATA(28,ISYS)
      FACTORP=SSDATA(29,ISYS)
      CALL SIZE(ILOC,ISYS,RATIO,FACTORW,FACTORV,FACTORP)
      E=SSDATA(30,ISYS)
      RETURN
      END
C
      SUBROUTINE SRBSUB(NWRITE)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
     * CLOTHW,
     * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMAT(3),
     * EFF(NWAUX)
      COMMON /FILE/ FNAME(NSYS)
```

```
      CHARACTER*12 FNAME,FILNAM(NRUM)
      CHARACTER*6 ADUMP
      CHARACTER*26 ALABEL(30)
      DATA ALABEL/ '  FLIGHT UNITS',' REDUNDANT UNITS',
     * '  AUXILIARY EQUIPMENT',
     * '  SPARES',' CONSUMABLES',' WATER(INITIAL STARTUP)',
C  ALABEL(7) NOT USED
     * '  ',' INITIAL LAUNCH WEIGHT',
     * ' RESUPPLY WEIGHT',' EQUIVALENT WEIGHT',' LIFETIME WEIGHT',
     * '  FLIGHT UNITS-POWER',' FLIGHT UNITS-HEAT REJ.',
     * '  AUXILIARY EQUIPMENT','MAINTENANCE TIME(HR)',
     * ' ON ORBIT VOLUME',
     * ' RESUPPLY VOLUME',' LIFETIME VOLUME',' RELIABILITY',
     * '  TECHNOLOGY READINESS',' PACING TECHNOLOGY PROB',
     * '  SAFETY',' MAINTAINABILITY',' COMMONALITY',' OVERALL',
     * '  MATERIAL LOST & STORED',' STORAGE-LOST MATERIAL',
     * '  ',' TOTAL POWER REQUIRED',
C ALABEL(28) NOT USED
     * '  MAKEUP WATER'/
      DO 5 I=1,NUMBER
      ISYS=ITYPE(I,1)
      DO 6 K=12,1,-1
    6 IF (FNAME(ISYS)(K:K).NE.' ') GOTO 7
    7 K2=K
      DO 8 K1=12,13-K,-1
      FILNAM(I)(K1:K1)=FNAME(ISYS)(K2:K2)
    8 K2=K2-1
      DO 9 K1=1,12-K
    9 FILNAM(I)(K1:K1)=' '
    5 CONTINUE
      NREAD=5
      N1=IFIX(N)
      NR=IFIX(R)
      DO 10 I=1,NUMBER
   10 IF (ITYPE(I,1).GE.31) GOTO 20
   20 IARS=I-1
      IWRS=I
      IF (IARS.EQ.0) GOTO 100
      IF (IARS.EQ.NUMBER) GOTO 75
      IFORM=2
      WRITE(NWRITE,3)
    3 FORMAT(4(/))
      WRITE(NWRITE,30) N1,NR,M
   30 FORMAT(T7,'OVERALL SUMMARY: ',I2,' CREW PERSONS, ',I3,' DAY ',
     * 'RESUPPLY, ',F4.1,' YEAR LIFETIME')
      WRITE(NWRITE,31)
   31 FORMAT(T5,'ITEM',T36,'ARS',T51,'WRS',T61,'TOTALS')
      WRITE(NWRITE,32)
   32 FORMAT(1X,'WEIGHTS(LB)')
      WRITE(NWRITE,50) ALABEL( 1),(TOTALS( 4,I),I=1,3)
   50 FORMAT(T2,A26,T25,<IFORM>(5X,F10.2),2X,F10.2)
      WRITE(NWRITE,50) ALABEL( 2),(TOTALS(24,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 3),(TOTALS(18,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(26),(TOTALS(21,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(27),(TOTALS(22,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 4),(TOTALS( 6,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 5),(TOTALS( 8,I),I=1,3)
      WRITE(NWRITE,55) ALABEL(30),AUXEQ(11,0,1),AUXEQ(11,0,1)
   55 FORMAT(T2,A26,T25,20X,F10.2,2X,F10.2)
      WRITE(NWRITE,50) ALABEL( 6),(TOTALS(26,I),I=1,3)
```

```
      WRITE(NWRITE,50) ALABEL( 8),(TOTALS(28,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 9),(TOTALS(30,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(10),(TOTALS(32,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(11),(TOTALS(33,I),I=1,3)
      WRITE(NWRITE,51)
   51 FORMAT(/,1X,'ORBITAL AVERAGE POWER/HEAT REJECTION(WATTS)')
      WRITE(NWRITE,50) ALABEL(12),(TOTALS(17,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(14),(TOTALS(20,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(29),(TOTALS(35,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(13),(TOTALS( 3,I),I=1,3)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
    1 FORMAT(I2)
      WRITE(NWRITE,3)
      WRITE(NWRITE,30) N1,NR,M
      WRITE(NWRITE,31)
      WRITE(NWRITE,58)
   58 FORMAT(1X,'VOLUMES(FT3)')
      WRITE(NWRITE,50) ALABEL( 1),(TOTALS( 5,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 2),(TOTALS(25,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 3),(TOTALS(19,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(27),(TOTALS(23,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 4),(TOTALS( 7,I),I=1,3)
      WRITE(NWRITE,50) ALABEL( 5),(TOTALS( 9,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(16),(TOTALS(29,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(17),(TOTALS(31,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(18),(TOTALS(34,I),I=1,3)
      WRITE(NWRITE,50) ALABEL(15),(TOTALS(16,I),I=1,3)
      WRITE(NWRITE,59)
   59 FORMAT(/,1X,'RATINGS(1 TO 8)')
      WRITE(NWRITE,50) ALABEL(19),(TOTALS(10,I),I=1,3)
   60 FORMAT(T2,A26,T24,<IFORM>(10X,F5.1),7X,F5.1)
      WRITE(NWRITE,60) ALABEL(20),(TOTALS(11,I),I=1,3)
      WRITE(NWRITE,60) ALABEL(21),(TOTALS(12,I),I=1,3)
      WRITE(NWRITE,60) ALABEL(22),(TOTALS(13,I),I=1,3)
      WRITE(NWRITE,60) ALABEL(23),(TOTALS(14,I),I=1,3)
      WRITE(NWRITE,60) ALABEL(24),(TOTALS(15,I),I=1,3)
      WRITE(NWRITE,60) ALABEL(25),(TOTALS(27,I),I=1,3)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
   75 IFORM =IARS
      WRITE(NWRITE,3)
      WRITE(NWRITE,61) N1,NR,M
   61 FORMAT(1X,'AIR REVITALIZATION SYSTEM: ',I2,' CREW PERSONS,',I3,
     * ' DAY RESUPPLY, ',F4.1,' YEAR LIFETIME')
      WRITE(NWRITE,62) (FILNAM(I),I=1,IARS)
   62 FORMAT(T5,'ITEM',T25,<IFORM>(3X,A12),6X,'TOTALS')
      WRITE(NWRITE,63) (SUBSYS(0,I),I=1,IARS)
   63 FORMAT(1X,'LOAD(LB/DAY)',T25,<IFORM>(8X,F7.2),/,1X,'WEIGHTS(LB)')
      WRITE(NWRITE,50) ALABEL( 1),(SUBSYS( 4,I),I=1,IARS),TOTALS( 4,1)
      WRITE(NWRITE,50) ALABEL( 2),(SUBSYS(24,I),I=1,IARS),TOTALS(24,1)
      WRITE(NWRITE,50) ALABEL( 3),(SUBSYS(18,I),I=1,IARS),TOTALS(18,1)
      WRITE(NWRITE,50) ALABEL(26),(SUBSYS(21,I),I=1,IARS),TOTALS(21,1)
      WRITE(NWRITE,50) ALABEL(27),(SUBSYS(22,I),I=1,IARS),TOTALS(22,1)
      WRITE(NWRITE,50) ALABEL( 4),(SUBSYS( 6,I),I=1,IARS),TOTALS( 6,1)
      WRITE(NWRITE,50) ALABEL( 5),(SUBSYS( 8,I),I=1,IARS),TOTALS( 8,1)
      WRITE(NWRITE,50) ALABEL( 6),(SUBSYS(26,I),I=1,IARS),TOTALS(26,1)
      WRITE(NWRITE,50) ALABEL( 8),(SUBSYS(28,I),I=1,IARS),TOTALS(28,1)
      WRITE(NWRITE,50) ALABEL( 9),(SUBSYS(30,I),I=1,IARS),TOTALS(30,1)
      WRITE(NWRITE,50) ALABEL(10),(SUBSYS(32,I),I=1,IARS),TOTALS(32,1)
      WRITE(NWRITE,50) ALABEL(11),(SUBSYS(33,I),I=1,IARS),TOTALS(33,1)
      WRITE(NWRITE,51)
```

```
      WRITE(NWRITE,50) ALABEL(12),(SUBSYS(17,I),I=1,IARS),TOTALS(17,1)
      WRITE(NWRITE,50) ALABEL(14),(SUBSYS(20,I),I=1,IARS),TOTALS(20,1)
      WRITE(NWRITE,50) ALABEL(29),(SUBSYS(35,I),I=1,IARS),TOTALS(35,1)
      WRITE(NWRITE,50) ALABEL(13),(SUBSYS( 3,I),I=1,IARS),TOTALS( 3,1)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
      WRITE(NWRITE,3)
      WRITE(NWRITE,61) N1,N,  M
      WRITE(NWRITE,62) (FILNAM(I),I=1,IARS)
      WRITE(NWRITE,71) (ITYPE(I,3),ITYPE(I,4),I=1,IARS)
   71 FORMAT(1X,'NUMBER OF UNITS/ALTERNATES',T25,3(10X,I2,'/',I2),
     * /,1X,'VOLUMES(FT3)')
      WRITE(NWRITE,50) ALABEL( 1),(SUBSYS( 5,I),I=1,IARS),TOTALS( 5,1)
      WRITE(NWRITE,50) ALABEL( 2),(SUBSYS(25,I),I=1,IARS),TOTALS(25,1)
      WRITE(NWRITE,50) ALABEL( 3),(SUBSYS(19,I),I=1,IARS),TOTALS(19,1)
      WRITE(NWRITE,50) ALABEL(27),(SUBSYS(23,I),I=1,IARS),TOTALS(23,1)
      WRITE(NWRITE,50) ALABEL( 4),(SUBSYS( 7,I),I=1,IARS),TOTALS( 7,1)
      WRITE(NWRITE,50) ALABEL( 5),(SUBSYS( 9,I),I=1,IARS),TOTALS( 9,1)
      WRITE(NWRITE,50) ALABEL(16),(SUBSYS(29,I),I=1,IARS),TOTALS(29,1)
      WRITE(NWRITE,50) ALABEL(17),(SUBSYS(31,I),I=1,IARS),TOTALS(31,1)
      WRITE(NWRITE,50) ALABEL(18),(SUBSYS(34,I),I=1,IARS),TOTALS(34,1)
      WRITE(NWRITE,50) ALABEL(15),(SUBSYS(16,I),I=1,IARS),TOTALS(16,1)
      WRITE(NWRITE,59)
      WRITE(NWRITE,60) ALABEL(19),(SUBSYS(10,I),I=1,IARS),TOTALS(10,1)
      WRITE(NWRITE,60) ALABEL(20),(SUBSYS(11,I),I=1,IARS),TOTALS(11,1)
      WRITE(NWRITE,60) ALABEL(21),(SUBSYS(12,I),I=1,IARS),TOTALS(12,1)
      WRITE(NWRITE,60) ALABEL(22),(SUBSYS(13,I),I=1,IARS),TOTALS(13,1)
      WRITE(NWRITE,60) ALABEL(23),(SUBSYS(14,I),I=1,IARS),TOTALS(14,1)
      WRITE(NWRITE,60) ALABEL(24),(SUBSYS(15,I),I=1,IARS),TOTALS(15,1)
      WRITE(NWRITE,60) ALABEL(25),(SUBSYS(27,I),I=1,IARS),TOTALS(27,1)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
      IF (IARS.EQ.NUMBER) GOTO 110
  100 IFORM=NUMBER-IARS
      IF (IFORM.GT.3) IFORM=3
      IPRNT1=IARS
   95 IPRNT1=IPRNT1+IFORM
      WRITE(NWRITE,3)
      WRITE(NWRITE,94) N1,NR,M
   94 FORMAT(1X,'WATER RECLAMATION SYSTEM:  ',I2,' CREW PERSONS,',I3,
     * ' DAY RESUPPLY, ',F4.1,' YEAR LIFETIME')
      WRITE(NWRITE,62) (FILNAM(I),I=IWRS,IPRNT1)
      WRITE(NWRITE,63) (SUBSYS(0,I),I=IWRS,IPRNT1)
   72 FORMAT(1X,'PROCESS EFFICIENCY',T25,<IFORM>(5X,F10.3),2X,F10.3)
      WRITE(NWRITE,50) ALABEL( 1),(SUBSYS( 4,I),I=IWRS,IPRNT1),
     * TOTALS( 4,2)
      WRITE(NWRITE,50) ALABEL( 2),(SUBSYS(24,I),I=IWRS,IPRNT1),
     * TOTALS(24,2)
      WRITE(NWRITE,50) ALABEL( 3),(SUBSYS(18,I),I=IWRS,IPRNT1),
     * TOTALS(18,2)
      WRITE(NWRITE,50) ALABEL(26),(SUBSYS(21,I),I=IWRS,IPRNT1),
     * TOTALS(21,2)
      WRITE(NWRITE,50) ALABEL(27),(SUBSYS(22,I),I=IWRS,IPRNT1),
     * TOTALS(22,2)
      WRITE(NWRITE,50) ALABEL( 4),(SUBSYS( 6,I),I=IWRS,IPRNT1),
     * TOTALS( 6,2)
      WRITE(NWRITE,50) ALABEL( 5),(SUBSYS( 8,I),I=IWRS,IPRNT1),
     * TOTALS( 8,2)
      WRITE(NWRITE,70) ALABEL(30),AUXEQ(11,0,1)
   70 FORMAT(T2,A26,T23,<IFORM>(15X),4X,F10.2)
      WRITE(NWRITE,50) ALABEL( 6),(SUBSYS(26,I),I=IWRS,IPRNT1),
     * TOTALS(26,2)
```

```
      WRITE(NWRITE,50) ALABEL( 8),(SUBSYS(28,I),I=IWRS,IPRNT1),
     * TOTALS(28,2)
      WRITE(NWRITE,50) ALABEL( 9),(SUBSYS(30,I),I=IWRS,IPRNT1),
     * TOTALS(30,2)
      WRITE(NWRITE,50) ALABEL(10),(SUBSYS(32,I),I=IWRS,IPRNT1),
     * TOTALS(32,2)
      WRITE(NWRITE,50) ALABEL(11),(SUBSYS(33,I),I=IWRS,IPRNT1),
     * TOTALS(33,2)
      WRITE(NWRITE,51)
      WRITE(NWRITE,50) ALABEL(12),(SUBSYS(17,I),I=IWRS,IPRNT1),
     * TOTALS(17,2)
      WRITE(NWRITE,50) ALABEL(14),(SUBSYS(20,I),I=IWRS,IPRNT1),
     * TOTALS(20,2)
      WRITE(NWRITE,50) ALABEL(13),(SUBSYS( 3,I),I=IWRS,IPRNT1),
     * TOTALS( 3,2)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
      WRITE(NWRITE,3)
      WRITE(NWRITE,94) N1,NR,M
        WRITE(NWRITE,62) (FILNAM(I),I=IWRS,IPRNT1)
        WRITE(NWRITE,72) (EFF(I),I=IWRS,IPRNT1)
        WRITE(NWRITE,71) (ITYPE(I,3),ITYPE(I,4),I=IWRS,IPRNT1)
      WRITE(NWRITE,30) ALABEL( 1),(SUBSYS( 5,I),I=IWRS,IPRNT1),
     * TOTALS( 5,2)
      WRITE(NWRITE,50) ALABEL( 2),(SUBSYS(25,I),I=IWRS,IPRNT1),
     * TOTALS(25,2)
      WRITE(NWRITE,50) ALABEL( 3),(SUBSYS(19,I),I=IWRS,IPRNT1),
     * TOTALS(19,2)
      WRITE(NWRITE,50) ALABEL(27),(SUBSYS(23,I),I=IWRS,IPRNT1),
     * TOTALS(23,2)
      WRITE(NWRITE,50) ALABEL( 4),(SUBSYS( 7,I),I=IWRS,IPRNT1),
     * TOTALS( 7,2)
      WRITE(NWRITE,50) ALABEL( 5),(SUBSYS( 9,I),I=IWRS,IPRNT1),
     * TOTALS( 9,2)
      WRITE(NWRITE,50) ALABEL(16),(SUBSYS(29,I),I=IWRS,IPRNT1),
     * TOTALS(29,2)
      WRITE(NWRITE,50) ALABEL(17),(SUBSYS(31,I),I=IWRS,IPRNT1),
     * TOTALS(31,2)
      WRITE(NWRITE,50) ALABEL(18),(SUBSYS(34,I),I=IWRS,IPRNT1),
     * TOTALS(34,2)
      WRITE(NWRITE,50) ALABEL(15),(SUBSYS(16,I),I=IWRS,IPRNT1),
     * TOTALS(16,2)
      WRITE(NWRITE,73)
   73 FORMAT(1X,'RATINGS(1 TO 8)')
      WRITE(NWRITE,60) ALABEL(19),(SUBSYS(10,I),I=IWRS,IPRNT1),
     * TOTALS(10,2)
      WRITE(NWRITE,60) ALABEL(20),(SUBSYS(11,I),I=IWRS,IPRNT1),
     * TOTALS(11,2)
      WRITE(NWRITE,60) ALABEL(21),(SUBSYS(12,I),I=IWRS,IPRNT1),
     * TOTALS(12,2)
      WRITE(NWRITE,60) ALABEL(22),(SUBSYS(13,I),I=IWRS,IPRNT1),
     * TOTALS(13,2)
      WRITE(NWRITE,60) ALABEL(23),(SUBSYS(14,I),I=IWRS,IPRNT1),
     * TOTALS(14,2)
      WRITE(NWRITE,60) ALABEL(24),(SUBSYS(15,I),I=IWRS,IPRNT1),
     * TOTALS(15,2)
      WRITE(NWRITE,60) ALABEL(25),(SUBSYS(27,I),I=IWRS,IPRNT1),
     * TOTALS(27,2)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
      IWRS=IPRNT1+1
      IFORM=NUMBER-IPRNT1
```

```
      IF (IFORM.EQ.0) GOTO 110
      IF (IFORM.GT.3) IFORM=3
      GOTO 95
  110 CONTINUE
      RETURN
      END
C
      SUBROUTINE WARSMAT(NWRITE)
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      CHARACTER*6 ADUMP
      NREAD=5
      N1=IFIX(N)
      WRITE(NWRITE,10) N1
   10 FORMAT(5(/),T15,'AIR REVITALIZATION SUMMARY (LB/DAY): ',I2,
     * ' CREW PERSONS')
      WRITE(NWRITE,20) PTOT,PPO2,PPN2
   20 FORMAT(T9,'TOTAL PRESSURE= ',F4.1,' PSIA   PPO2= ',F4.1,
     * ' PSIA   PPN2= ',F4.1,' PSIA')
      WRITE(NWRITE,30)
   30 FORMAT(T4,'O2 REQUIREMENTS',T26,'H2 REQUIREMENTS',T49,
     * 'CO2 PRODUCED',T66,'N2 REQUIREMENTS')
      WRITE(NWRITE,40) O2MET,H2EDC,CO2MET,N2LEAK
   40 FORMAT(1X,'METABOLIC  = ',F5.2,T24,'EDC',T35,'= ',F5.2,T46,
     * 'METABOLIC= ',F5.2,T65,'MOD LEAK = ',F5.2)
      WRITE(NWRITE,50) O2EVA,H2BSCH,CO2EVA,N2AIR
   50 FORMAT(1X,'EVA',T13,'= ',F5.2,T24,'BOSCH',T35,'= ',F5.2,T46,
     * 'EVA',T55,'= ',F5.2,T65,'AIRLOCK  = ',F5.2)
      WRITE(NWRITE,60) O2LEAK,H2SAB,N2RPRS
   60 FORMAT(1X,'MOD LEAKAGE= ',F5.2,T24,'SABATIER   = ',F5.2,T65,
     * 'MOD REPRS= ',F5.2)
      WRITE(NWRITE,70) O2AIR,N2COM
   70 FORMAT(1X,'AIRLOCK    = ',F5.2,T65,'COMMODE  = ',F5.2)
      WRITE(NWRITE,80) O2RPRS,N2PURG
   80 FORMAT(1X,'MOD REPRESS= ',F5.2,T65,'PURGE    = ',F5.2)
      WRITE(NWRITE,90) O2COM
   90 FORMAT(1X,'COMMODE    = ',F5.2)
```

```
      WRITE(NWRITE,100) O2EDC
  100 FORMAT(1X,'EDC',T13,'= ',F5.2)
      WRITE(NWRITE,110) O2TOT,H2REQ,CO2TOT,N2TOT
  110 FORMAT(T4,'TOTAL    = ',F5.2,T26,'TOTAL    = ',F5.2,T48,
     * 'TOTAL  = ',F5.2,T67,'TOTAL  = ',F5.2)
      WRITE(NWRITE,120)
  120 FORMAT()
      WRITE(NWRITE,130)
  130 FORMAT(T16,'O2 GENERATION',T38,'CO2 REMOVAL',T61,'CO2 REDUCTION')
      WRITE(NWRITE,140) H2OGEN,H2OREMRQ,CO2TOT
  140 FORMAT(1X,'INPUTS:',T12,'H2O REQUIRED= ',F5.2,T35,
     * 'H2O CONSUMED= ',F5.2,T58,'CO2 AVAILABLE= ',F5.2)
      WRITE(NWRITE,150) O2EDC,H2RED
  150 FORMAT(T35,'O2  CONSUMED= ',F5.2,T58,'H2 CONSUMED  = ',F5.2)
      WRITE(NWRITE,160) H2EDC
  160 FORMAT(T35,'H2  CONSUMED= ',F5.2)
      WRITE(NWRITE,170)
  170 FORMAT()
      WRITE(NWRITE,180) O2TOT,CO2MET,H2ORED
  180 FORMAT(1X,'OUTPUTS:  O2 GENERATED= ',F5.2,T35,'CO2 REMOVED = ',
     * F5.2,T58,'H2O PRODUCED = ',F5.2)
      WRITE(NWRITE,190) H2TOT,H2OREM,C
  190 FORMAT(T12,'H2 GENERATED= ',F5.2,T35,'H2O LATENT  = ',F5.2,
     * T58,'CARBON STORED= ',F5.2)
      WRITE(NWRITE,200) ADUMP,H2STOR
  200 FORMAT(T58,'H2  ',A6,'   = ',F5.2)
      WRITE(NWRITE,210) ADUMP,CH4
  210 FORMAT(T58,'CH4 ',A6,'   = ',F5.2)
      WRITE(NWRITE,220) ADUMP,CO2LST
  220 FORMAT(T58,'CO2 ',A6,'   = ',F5.2)
      WRITE(NWRITE,230) H2OARS,CO2RED
  230 FORMAT(1X,'H2O LOST BY ARS= ',F6.2,T58,'CO2 REDUCED  = ',F5.2)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
    1 FORMAT(I2)
      RETURN
      END
C
      SUBROUTINE WWRSMAT(NWRITE)
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWP,
```

```
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      CHARACTER*6 ADUMP
      NREAD=5
      N1=IFIX(N)
      WRITE(NWRITE,10) N1
   10 FORMAT(5(/),T16,'WATER RECLAMATION SUMMARY (LB/DAY): ',I2,
     * ' CREW PERSONS')
      WRITE(NWRITE,20)
   20 FORMAT(1X,'H2O REQUIREMENTS:',T47,'LOAD    RECOVERED',
     * '   POTABLE    LOST')
      WRITE(NWRITE,30) HAND,COND,CONDR,CONDP,CONDS
   30 FORMAT(T4,'HAND WASH    =',F6.2,T27,'1.CONDENSATE',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,40) SHOWER,SWEAT
   40 FORMAT(T4,'SHOWER',T17,'=',F6.2,T30,'SWEAT & RESP',T46,F6.2)
      WRITE(NWRITE,50) CLOTHES,HYGIENL
   50 FORMAT(T4,'CLOTHES WASH =',F6.2,T30,'HYGIENE LATENT',T46,F6.2)
      WRITE(NWRITE,60) DISHW,FOODPL
   60 FORMAT(T4,'DISHWASH     =',F6.2,T30,'FOOD PREP LAT',T46,F6.2)
      WRITE(NWRITE,70) DRINK,LAUNL
   70 FORMAT(T4,'DRINKING     =',F6.2,T30,'LAUNDRY LATENT',T46,F6.2)
      WRITE(NWRITE,80) FOODP,DISHWL
   80 FORMAT(T4,'FOOD PREP    =',F6.2,T30,'DISHWASH LAT',T46,F6.2)
      WRITE(NWRITE,90) URINFL,EXPH2OL
   90 FORMAT(T4,'URINAL FLUSH =',F6.2,T30,'EXPERIMENT LAT',T46,F6.2)
      WRITE(NWRITE,100) EXPH2O,H2OREM
  100 FORMAT(T4,'EXPERIMENT  =',F6.2,T30,'CO2 REMOVAL',T46,F6.2)
      WRITE(NWRITE,110) EVAH2O,REDH2O,REDH2OR,REDH2OP,REDH2OS
  110 FORMAT(T4,'EVA',T17,'=',F6.2,T27,'2.CO2 REDUCTION',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,120) H2OREMRQ,HYGIENLD,HYGIENR,HYGIENP,HYGIENS
  120 FORMAT(T4,'CO2 REMOVAL =',F6.2,T27,'3.HYGIENE H2O',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,130) H2OGEN,HANDLD
  130 FORMAT(T4,'O2 GEN',T17,'=',F6.2,T30,'HAND WASH',T46,F6.2)
      WRITE(NWRITE,140) POTREQ,SHOWLD
  140 FORMAT(T3,'POTABLE REQD  =',F6.2,T30,'SHOWER H2O',T46,F6.2)
      WRITE(NWRITE,150) H2OREQ,CLOTHLD,CLOTHR,CLOTHP,CLOTHS
  150 FORMAT(1X,'TOTAL H2O REQD =',F6.2,T27,'4.CLOTHES WASH',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,160) URINLD,URINR,URINP,URINS
  160 FORMAT(T27,'5.URINE H2O',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,170) H2OMET,H2OURN
  170 FORMAT(T4,'METABOLIC   =',F6.2,T30,'H2O IN URINE',T46,F6.2)
      WRITE(NWRITE,180) H2OFOOD,URINFL
  180 FORMAT(T4,'H2O IN FOOD  =',F6.2,T30,'URINAL FLUSH',T46,F6.2)
      WRITE(NWRITE,190) H2OPROD,EVAH2OLD
  190 FORMAT(1X,'H2O PRODUCTION =',F6.2,T30,'EVA/EXP H2O',T46,F6.2)
      WRITE(NWRITE,200) H2OTOT,DISHWLD,DISHWR,DISHWP,DISHWS
  200 FORMAT(1X,'TOTAL H2O AVAIL=',F6.2,T27,'6.DISHWASH H2O',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      WRITE(NWRITE,210) H2OARS,OTHERLD,OTHERR,OTHERP,OTHERS
  210 FORMAT(T4,'H2O LOST(ARS)=',F6.2,T29,'EVA/EXP LOST',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
      ADUM=0.
      WRITE(NWRITE,220) FECAL,ADUM,ADUM,FECAL
  220 FORMAT(T29,'FECAL H2O',T46,F6.2,
```

```
     * T55,F6.2,T66,F6.2,T75,F6.2)
       WRITE(NWRITE,230) H2ONET,H2OLOAD,H2OREC,POTREC,H2OSTOR
  230 FORMAT(1X,'NET H2O LOAD   =',F6.2,T27,'TOTALS',T46,F6.2,
     * T56,F6.2,T66,F6.2,T75,F6.2)
       IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
     1 FORMAT(I2)
       RETURN
       END
C
       SUBROUTINE RESUP
       REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
      * LAUNL
       PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
       PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
       PARAMETER (NCOMPAR=20)
       COMMON /CHAR/ ADUMP
       COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
      * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
      * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
      * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
      * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
      * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
      * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
       COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
      * O2COM,O2EDC,O2TOT,H2EDC,H2BSCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
      * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
      * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
      * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
      * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
      * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
      * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
      * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
      * DISHWS,OTHERLD,OTHERR,OTHERP,OTHERS,H2OLOAD,H2OREC,POTREC,POTREQ,
      * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
      * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
       COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
      * CLOTHW,
      * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMAT(3),
      * EFF(NWAUX)
       CHARACTER*6 ADUMP
C
C  AUXEQ(I,J, 1) = WEIGHT (LBS)
C  AUXEQ(I,J, 2) = VOLUME (FT3)
C  AUXEQ(I,J, 3) = POWER (WATTS)
C
C  AUXEQ(I, 0,1) = WATER WEIGHT(INITIAL STARTUP)
C  AUXEQ(I, 1,K) = AUXILIARY EQUIPMENT, EXCLUDING TANKAGE
C  AUXEQ(I, 2,K) = TANKAGE
C  AUXEQ(I, 3,K) = MATERIAL (WEIGHT ONLY): CO2,H2,OR H2O
C  AUXEQ(I, 4,K) = MATERIAL (WEIGHT ONLY): CH4 OR C
C  AUXEQ(I, 5,K) = STORAGE EQUIPMENT FOR CO2,H2,OR H2O
C  AUXEQ(I, 6,K) = STORAGE EQUIPMENT FOR CH4 OR C
C  AUXEQ(I, 7,K) = TOTAL
C
C  AUXEQ( 1,J,K) = O2 GENERATION SUBSYSTEM
C  AUXEQ( 2,J,K) = CO2 REMOVAL SUBSYSTEM
C  AUXEQ( 3,J,K) = CO2 REDUCTION SUBSYSTEM
C  AUXEQ( 4,J,K) = WATER RECLAMATION SUBSYSTEM # 1
C  AUXEQ( 5,J,K) = WATER RECLAMATION SUBSYSTEM # 2
C  AUXEQ( 6,J,K) = WATER RECLAMATION SUBSYSTEM # 3
```

```
C  AUXEQ( 7,J,K) = WATER RECLAMATION SUBSYSTEM # 4
C  AUXEQ( 8,J,K) = WATER RECLAMATION SUBSYSTEM # 5
C  AUXEQ( 9,J,K) = WATER RECLAMATION SUBSYSTEM # 6
C  AUXEQ(10,J,K) = EVA/EXPERIMENT H2O LOST
C  AUXEQ(11,J,K) = MAKEUP H2O
C  AUXEQ(12,J,K) = EXCESS H2O
C  AUXEQ(13,J,K) = RESUPPLY CLOTHES
C  AUXEQ(14,J,K) = RESUPPLY (DISPOSABLE) DISHES
C  AUXEQ(15,J,K) = TRASH
C  AUXEQ(16,J,K) = TOTALS FOR ARS (1-3)
C  AUXEQ(17,J,K) = TOTALS FOR WRS (4-12)
C  AUXEQ(18,J,K) = GRAND TOTALS (1-15)
C
      DO 6 I=1,18
      DO 5 J=0,7
      DO 4 K=1,3
    4 AUXEQ(I,J,K)=0.
    5 CONTINUE
    6 CONTINUE
      ICNTW=3
      DO 100 I=1,NUMBER
      ISYS=ITYPE(I,1)
      IF (ISYS.GE.21.AND.ISYS.LE.30) THEN
        DO 10 K=1,3
        DO 12 J=1,NAAUX
   12   AUXEQ(1,1,K)=AUXEQ(1,1,K)+AUXILA(I,J,K)*AUXILA(I,J,0)
        DO 11 J=1,6
   11   AUXEQ(1,7,K)=AUXEQ(1,7,K)+AUXEQ(1,J,K)
   10   CONTINUE
      ENDIF
      IF (ISYS.GE.1.AND.ISYS.LE.10) THEN
        DO 20 K=1,3
        DO 22 J=1,NAAUX
   22   AUXEQ(2,1,K)=AUXEQ(2,1,K)+AUXILA(I,J,K)*AUXILA(I,J,0)
        DO 21 J=1,6
   21   AUXEQ(2,7,K)=AUXEQ(2,7,K)+AUXEQ(2,J,K)
   20   CONTINUE
      ENDIF
      IF (ISYS.GE.11.AND.ISYS.LE.20) THEN
        AMLTFC=2.-FLOAT(ITYPE(I,2))
        DO 30 K=1,3
        DO 31 J=1,NAAUX
        IF (J.LT.6.OR.J.GT.10) AUXEQ(3,1,K)=AUXEQ(3,1,K)+AUXILA(I,J,0)*
     *   AUXILA(I,J,K)
        IF (J.EQ.6.OR.J.EQ.8) AUXEQ(3,1,K)=AUXEQ(3,1,K)+AUXILA(I,J,0)*
     *   AUXILA(I,J,K)*AMLTFC
   31   CONTINUE
   30   CONTINUE
        IF (ISYS.EQ.11) THEN
          AUXEQ(3,3,1)=H2STORR
          AUXEQ(3,4,1)=CRES
          DO 33 K=1,3
          AUXEQ(3,5,K)=AUXILA(I,7,0)*AUXILA(I,7,K)*AMLTFC
          AUXEQ(3,6,K)=AUXILA(I,10,0)*AUXILA(I,10,K)
   33     CONTINUE
        ENDIF
        IF (ISYS.EQ.12) THEN
          AUXEQ(3,3,1)=CO2LSTR
          AUXEQ(3,4,1)=CH4R
          RATIO=CH4R/(CH4R+CO2LSTR)
```

```
            DO 34 K=1,3
            QUANT=AUXILA(I,9,0)*AUXILA(I,9,K)*AMLTFC
            AUXEQ(3,5,K)=(1.-RATIO)*QUANT
            AUXEQ(3,6,K)=RATIO*QUANT
   34       CONTINUE
         ENDIF
         DO 36 K=1,3
         IF (ISYS.EQ.12) THEN
           AUXMAT(K)=(AUXEQ(3,3,K)+AUXEQ(3,4,K))*AMLTFC
         ELSE
           AUXMAT(K)=AUXEQ(3,3,K)*AMLTFC+AUXEQ(3,4,K)
         ENDIF
         AUXEQ(3,7,K)=AUXEQ(3,1,K)+AUXEQ(3,2,K)+AUXMAT(K)+AUXEQ(3,5,K)+
     *      AUXEQ(3,6,K)
   36    CONTINUE
      ENDIF
      IF (ISYS.GE.31) THEN
        ICNTW=ICNTW+1
        AUXEQ(ICNTW,0,1)=SUBSYS(0,I)*R/(AUXILW(I,0,0)+1.)
        AUXEQ(ICNTW,3,1)=SUBSYS(0,I)*(1.-EFF(I))*R
        DO 40 K=1,3
        DO 42 J=2,NWAUX
   42   AUXEQ(ICNTW,1,K)=AUXEQ(ICNTW,1,K)+AUXILW(I,J,K)*AUXILW(I,J,0)
        AUXEQ(ICNTW,2,K)=AUXEQ(ICNTW,0,1)*AUXILW(I,1,0)*AUXILW(I,1,K)
C  ASSUMES 50% BRINE
        AUXEQ(ICNTW,5,K)=(AUXEQ(ICNTW,3,1)*2.)*AUXILW(I,1,K)
        DO 41 J=0,6
   41   AUXEQ(ICNTW,7,K)=AUXEQ(ICNTW,7,K)+AUXEQ(ICNTW,J,K)
   40   CONTINUE
      ENDIF
  100 CONTINUE
C
C  TANKS FOR STORED,MAKEUP,EXCESS,EVA/EXP LOST H2O SIZED USING DATABASE VALUES
C
      STOREW=0.
      DO 45 I=1,NWNUM
      IF (IWATER(I).NE.0) GOTO 45
      IF (I.EQ.1) STOREW=STOREW+COND
      IF (I.EQ.2) STOREW=STOREW+REDH2O
      IF (I.EQ.3) STOREW=STOREW+HYGIENLD
      IF (I.EQ.4) STOREW=STOREW+CLOTHLD
      IF (I.EQ.5) STOREW=STOREW+URINLD
      IF (I.EQ.6) STOREW=STOREW+FLSHWLD
   45 CONTINUE
      IF (STOREW.EQ.0.) GOTO 49
      ICNTW=ICNTW+1
      AUXEQ(ICNTW,3,1)=STOREW*R
      DO 46 K=1,3
      AUXEQ(ICNTW,5,K)=AUXEQ(ICNTW,3,1)*WRSAUX(1,K)
      AUXEQ(ICNTW,7,K)=AUXEQ(ICNTW,3,K)+AUXEQ(ICNTW,5,K)
   46 CONTINUE
   49 AUXEQ(10,3,1)= OTHERS*R
      AUXEQ(11,0,1)= MAKEUPR
      AUXEQ(12,3,1)= EXCESSR
      DO 50 K=1,3
      AUXEQ(10,5,K)= AUXEQ(10,3,1)*WRSAUX(1,K)
      AUXEQ(10,7,K)= AUXEQ(10,3,K)+AUXEQ(10,5,K)
      AUXEQ(11,5,K)= AUXEQ(11,0,1)*WRSAUX(1,K)
      AUXEQ(11,7,K)= AUXEQ(11,0,K)+AUXEQ(11,5,K)
      AUXEQ(12,5,K)= AUXEQ(12,3,1)*WRSAUX(1,K)
```

```
      AUXEQ(12,7,K)= AUXEQ(12,3,K)+AUXEQ(12,5,K)
   50 CONTINUE
      CLOTHW=DESIGN(34)*N
      CLOTHV=DESIGN(35)*N
      DAYS=DESIGN(36)
      IF (DESIGN(30).EQ.0.) DAYS=R
      AUXEQ(13,1,1)= DESIGN(46)
      AUXEQ(13,1,2)= DESIGN(47)
      AUXEQ(13,1,3)= DESIGN(48)
      AUXEQ(13,4,1)= CLOTHW*DAYS
      AUXEQ(13,6,2)= CLOTHV*DAYS
      AUXEQ(13,7,1)= AUXEQ(13,4,1)+AUXEQ(13,1,1)
      AUXEQ(13,7,2)= AUXEQ(13,6,2)+AUXEQ(13,1,2)
      AUXEQ(13,7,3)= AUXEQ(13,1,3)
C
      DISHWGHT=0.
      DISHV=0.
      AUXEQ(14,1,1)=DESIGN(49)
      AUXEQ(14,1,2)=DESIGN(50)
      AUXEQ(14,1,3)=DESIGN(51)
      IF (DESIGN(31).EQ.0.) THEN
        DISHWGHT=DESIGN(38)*N
        DISHV=DESIGN(39)*N
        AUXEQ(14,4,1)= DISHWGHT*R
        AUXEQ(14,6,2)= DISHV*R
      ENDIF
      AUXEQ(14,7,1)= AUXEQ(14,4,1)+AUXEQ(14,1,1)
      AUXEQ(14,7,2)= AUXEQ(14,6,2)+AUXEQ(14,1,2)
      AUXEQ(14,7,3)= AUXEQ(14,1,3)
C
      TRASHW=DESIGN(40)*N+DISHWGHT
      AUXEQ(15,1,1)=DESIGN(43)
      AUXEQ(15,1,2)=DESIGN(44)
      AUXEQ(15,1,3)=DESIGN(45)
      AUXEQ(15,4,1)=TRASHW*R
      RATIO=1.0
      IF (DESIGN(43).GT.0.) RATIO=DESIGN(42)
      TRASHV=RATIO*(DESIGN(41)*N+DISHV)
      AUXEQ(15,6,2)=TRASHV*R
      AUXEQ(15,7,1)=AUXEQ(15,4,1)+AUXEQ(15,1,1)
      AUXEQ(15,7,2)=AUXEQ(15,6,2)+AUXEQ(15,1,2)
      AUXEQ(15,7,3)=AUXEQ(15,1,3)
C
      DO 60 K=1,3
      AUXEQ(16,0,K)=AUXEQ(1,0,K)+AUXEQ(2,0,K)+AUXEQ(3,0,K)
      AUXEQ(16,1,K)=AUXEQ(1,1,K)+AUXEQ(2,1,K)+AUXEQ(3,1,K)
      AUXEQ(16,2,K)=AUXEQ(1,2,K)+AUXEQ(2,2,K)+AUXEQ(3,2,K)
      AUXEQ(16,4,K)=AUXEQ(1,3,K)+AUXEQ(1,4,K)+AUXEQ(2,3,K)+
     * AUXEQ(2,4,K)+AUXMAT(K)
      AUXEQ(16,6,K)=AUXEQ(1,5,K)+AUXEQ(1,6,K)+AUXEQ(2,5,K)+
     * AUXEQ(2,6,K)+AUXEQ(3,5,K)+AUXEQ(3,6,K)
      AUXEQ(16,7,K)=AUXEQ(1,7,K)+AUXEQ(2,7,K)+AUXEQ(3,7,K)
      DO 61 J=0,7
      DO 62 II=4,ICNTW
   62 AUXEQ(17,J,K)=AUXEQ(17,J,K)+AUXEQ(II,J,K)
C INITIAL WATER LOAD
      AUXEQ(17,0,1)=H2OLOAD-OTHERLD-FECAL
      AUXEQ(17,J,K)=AUXEQ(17,J,K)+AUXEQ(10,J,K)+AUXEQ(11,J,K)+
     * AUXEQ(12,J,K)
   61 CONTINUE
```

```
      AUXEQ(17,0,K)=AUXEQ(17,0,K)+AUXEQ(13,0,K)+AUXEQ(14,0,K)+
     * AUXEQ(15,0,K)
      AUXEQ(17,1,K)=AUXEQ(17,1,K)+AUXEQ(13,1,K)+AUXEQ(14,1,K)+
     * AUXEQ(15,1,K)
      AUXEQ(17,2,K)=AUXEQ(17,2,K)+AUXEQ(13,2,K)+AUXEQ(14,2,K)+
     * AUXEQ(15,2,K)
      AUXEQ(17,3,K)=AUXEQ(17,3,K)+AUXEQ(13,4,K)+AUXEQ(14,4,K)+
     * AUXEQ(15,4,K)
      AUXEQ(17,5,K)=AUXEQ(17,5,K)+AUXEQ(13,6,K)+AUXEQ(14,6,K)+
     * AUXEQ(15,6,K)
      AUXEQ(17,7,K)=AUXEQ(17,7,K)+AUXEQ(13,7,K)+AUXEQ(14,7,K)+
     * AUXEQ(15,7,K)
      AUXEQ(18,0,K)=AUXEQ(16,0,K)+AUXEQ(17,0,K)
      AUXEQ(18,1,K)=AUXEQ(16,1,K)+AUXEQ(17,1,K)
      AUXEQ(18,2,K)=AUXEQ(16,2,K)+AUXEQ(17,2,K)
      AUXEQ(18,4,K)=AUXEQ(16,4,K)+AUXEQ(17,3,K)
      AUXEQ(18,6,K)=AUXEQ(16,6,K)+AUXEQ(17,5,K)
      AUXEQ(18,7,K)=AUXEQ(16,7,K)+AUXEQ(17,7,K)
   60 CONTINUE
      RETURN
      END
C
      SUBROUTINE DESGN8
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C N,NUMBER OF CREW
      DESIGN( 1)=6.
C R,RESUPPLY INTERVAL(DAYS)
      DESIGN( 2)=90.
C M,MISSION DURATION(YEARS)
      DESIGN( 3)=10.
C N2 CABIN PARTIAL PRESSURE(PSIA)
      DESIGN( 4)=11.50
C O2 CABIN PARTIAL PRESSURE(PSIA)
      DESIGN( 5)=3.20
C CABIN LEAKAGE(LB/DAY)
      DESIGN( 6)=5.0
C AIR LOCK GAS LOSS(LB/DAY)
      DESIGN( 7)=3.0
C METABOLIC O2(LB/MAN-DAY)
      DESIGN( 8)=1.84
C METABOLIC CO2(LB/MAN-DAY)
      DESIGN( 9)=2.20
C RE,EVA MAN-HOURS/RESUPPLY INTERVAL(1029.)
      DESIGN(10)=0.0
C EVA O2(LB/MAN-HOUR)
      DESIGN(11)=0.153
C EVA CO2(LB/MAN-HOUR)
      DESIGN(12)=0.185
```

```
C EVA H2O(LB/MAN-HOUR) (1.21)
      DESIGN(13)=0.00
C EVA WATER LOST(LB/MAN-HOUR) (0.25)
      DESIGN(14)=0.00
C DRINKING H2O(LB/MAN-DAY)
      DESIGN(15)=4.09
C FOOD PREP H2O(LB/MAN-DAY)
      DESIGN(16)=1.64
C H2O IN FOOD(LB/MAN-DAY)
      DESIGN(17)=1.10
C METABOLIC H2O PROD(LB/MAN-DAY)
      DESIGN(18)=0.76
C HYGIENE LATENT H2O(LB/MAN-DAY)
      DESIGN(19)=0.96
C FOOD PREP LATENT H2O(LB/MAN-DAY)
      DESIGN(20)=0.06
C EXPERIMENTS LATENT H2O(LB/MAN-DAY) (1.0)
      DESIGN(21)=0.00
C LAUNDRY LATENT H2O(LB/MAN-DAY)
      DESIGN(22)=0.13
C DISHWASHER LATENT H2O(LB/MAN-DAY)
      DESIGN(23)=0.036
C SWEAT & RESP H2O(LB/MAN-DAY)
      DESIGN(24)=4.02
C URINE H2O(LB/MAN-DAY)
      DESIGN(25)=3.31
C URINAL FLUSH H2O(LB/MAN-DAY)
      DESIGN(26)=1.09
C FECAL H2O(LB/MAN-DAY)
      DESIGN(27)=0.20
C HAND WASH H2O(LB/MAN-DAY)
      DESIGN(28)=4.00
C SHOWER H2O(LB/MAN-DAY)
      DESIGN(29)=8.00
C CLOTHES WASH H2O(LB/MAN-DAY)
      DESIGN(30)=27.50
C DISHWASHER H2O(LB/MAN-DAY)
      DESIGN(31)=7.63
C EXPERIMENT H2O(LB/MAN-DAY) (1.0)
      DESIGN(32)=0.00
C EXPERIMENT H2O LOST(LB/MAN-DAY)
      DESIGN(33)=0.00
C CLOTHING WEIGHT(LB/MAN-DAY) (2.19)
      DESIGN(34)=0.00
C CLOTHING VOLUME(FT3/MAN-DAY) (0.72)
      DESIGN(35)=0.00
C RESUPPLY CLOTHING WITH CLOTHES WASH(DAYS) (14)
      DESIGN(36)=14.0
C UNUSED
      DESIGN(37)=0.0
C RESUPPLY DISHES W/O DISHWASH(LB/MAN-DAY)
      DESIGN(38)=0.00
C RESUPPLY DISHES W/O DISHWASH(FT3/MAN-DAY)
      DESIGN(39)=0.00
C TRASH WEIGHT(LB/MAN-DAY) (1.0)
      DESIGN(40)=0.00
C TRASH VOLUME(FT3/MAN-DAY) (0.1)
      DESIGN(41)=0.00
C TRASH COMPACTION RATIO(FRACTION OF ORIGINAL VOL)
      DESIGN(42)=0.00
```

```
C TRASH COMPACTOR WEIGHT(LB)
      DESIGN(43)=0.00
C TRASH COMPACTOR VOLUME(FT3)
      DESIGN(44)=0.00
C TRASH COMPACTOR POWER(WATTS)
      DESIGN(45)=0.00
C CLOTHES WASHER WEIGHT(LB)
      DESIGN(46)=0.00
C CLOTHES WASHER VOLUME(FT3) (12.)
      DESIGN(47)=0.00
C CLOTHES WASHER POWER(WATTS)
      DESIGN(48)=0.00
C DISHWASHER WEIGHT(LB) (100.)
      DESIGN(49)=0.00
C DISHWASHER VOLUME(FT3) (12.)
      DESIGN(50)=0.00
C DISHWASHER POWER(WATTS) (1200.)
      DESIGN(51)=0.00
C POWER PENALTY,LIGHT SIDE(LB/WATT)
      DESIGN(52)=0.371
C POWER PENALTY,DARK SIDE(LB/WATT)
      DESIGN(53)=1.366
C AVG HEAT REJECTION PENALTY(LB/WATT)
      DESIGN(54)=0.200
C TIME IN SUNLIGHT PER ORBIT(MIN)
      DESIGN(55)=60.0
C TIME IN DARKNESS PER ORBIT(MIN)
      DESIGN(56)=34.5
      RETURN
      END
C
      SUBROUTINE SSDATS
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /FILE/ FNAME(NSYS)
      CHARACTER*12 FNAME
      CHARACTER*6 ADUMP
C
C   KEY
C       ISYS=NO. OF SUBSYSTEM (1-50)
C
C  SSDATA( 1,ISYS)= AVERAGE POWER REQUIRED,DAYLIGHT SIDE OF ORBIT(WATTS)
C  SSDATA( 2,ISYS)= AVERAGE POWER REQUIRED,NIGHT SIDE OF ORBIT(WATTS)
C  SSDATA( 3,ISYS)= HEAT REJECTION (WATTS)
C  SSDATA( 4,ISYS)= INITIAL FLIGHT UNIT (LB)
C  SSDATA( 5,ISYS)= INITIAL FLIGHT UNIT (FT3)
C  SSDATA( 6,ISYS)= SPARES OVER RESUPPLY INTERVAL (LB)
C  SSDATA( 7,ISYS)= SPARES OVER RESUPPLY INTERVAL (FT3)
C  SSDATA( 8,ISYS)= CONSUMMABLES OVER RESUPPLY INTERVAL (LB)
C  SSDATA( 9,ISYS)= CONSUMMABLES OVER RESUPPLY INTERVAL (FT3)
```

```
C  SSDATA(10,ISYS)= RELIABILITY (1 TO 8)
C  SSDATA(11,ISYS)= TECHNOLOGY READINESS (1 TO 8)
C  SSDATA(12,ISYS)= PACING TECHNOLOGY PROBLEMS (1 TO 8)
C  SSDATA(13,ISYS)= SAFETY INDEX (1 TO 8)
C  SSDATA(14,ISYS)= MAINTAINABILITY INDEX (1 TO 8)
C  SSDATA(15,ISYS)= COMMONALITY INDEX (1 TO 8)
C  SSDATA(16,ISYS)= MAINTENANCE TIME OVER RESUPPLY INTERVAL (HR)
C  SSDATA(17,ISYS)= UNUSED
C  SSDATA(18,ISYS)= LB/DAY O2 USED TO SIZE SYSTEM (O2 GEN)
C  SSDATA(19,ISYS)= LB/DAY CO2 USED TO SIZE SYSTEM (CO2REM,CO2RED)
C  SSDATA(20,ISYS)= LB/DAY H2O USED TO SIZE SYSTEM (WRS)
C  SSDATA(21,ISYS)= H2 USED OVER RESUPPLY INTERVAL (LBM)-EDC,CO2RED
C  SSDATA(22,ISYS)= O2 USED OVER RESUPPLY INTERVAL (LBM)-EDC
C  SSDATA(23,ISYS)= H2O PRODUCED OVER RESUPPLY INTERVAL(LBM)-CO2RED,CO2REM
C  SSDATA(24,ISYS)= C PRODUCED OVER RESUPPLY INTERVAL(LBM)-BOSCH,SAB+CH4 DIS
C  SSDATA(25,ISYS)= CH4 PRODUCED OVER RESUPPLY INTERVAL(LBM)-SABATIER
C  SSDATA(26,ISYS)= CO2 NOT REACTED OVER RESUPPLY INTERVAL(LBM)-SABATIER
C  SSDATA(27,ISYS)= SIZING FACTOR FOR WEIGHTS
C  SSDATA(28,ISYS)= SIZING FACTOR FOR VOLUMES
C  SSDATA(29,ISYS)= SIZING FACTOR FOR POWERS
C  SSDATA(30,ISYS)= E, PROCESS EFFICIENCY
C
C  SIZING FACTOR= (DELTA SIZE/REF SIZE) / ((DELTA LB/DAY)/(REF LB/DAY))
C  NOTE:  CHANGING REFERENCE SIZE CHANGES SIZING FACTOR
C
C  O2 GENERATION
C
C  21. STATIC FEED-WES
C
      ISYS=21
      FNAME(ISYS)='SF-WES'
      SSDATA( 1,ISYS)= 1150.
      SSDATA( 2,ISYS)= 1150.
      SSDATA( 3,ISYS)= 133.
      SSDATA( 4,ISYS)= 118.
      SSDATA( 5,ISYS)= 1.6
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 6.
      SSDATA(11,ISYS)= 6.
      SSDATA(12,ISYS)= 6.
      SSDATA(13,ISYS)= 6.
      SSDATA(14,ISYS)= 6.
      SSDATA(15,ISYS)= 4.
      SSDATA(16,ISYS)= 0.3
      SSDATA(18,ISYS)= 11.04
      SSDATA(27,ISYS)= 0.666
      SSDATA(28,ISYS)= 0.685
      SSDATA(29,ISYS)= 0.952
      SSDATA(30,ISYS)= 1.0
C
C  22. SOLID POLYMER-LIQUID FEED
C
      ISYS=22
      FNAME(ISYS)='SOLID POL-LF'
      SSDATA( 1,ISYS)= 1330.
      SSDATA( 2,ISYS)= 1330.
      SSDATA( 3,ISYS)= 200.
```

```
      SSDATA( 4,ISYS)= 108.
      SSDATA( 5,ISYS)= 2.55
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 4.0
      SSDATA(11,ISYS)= 5.0
      SSDATA(12,ISYS)= 1.0
      SSDATA(13,ISYS)= 6.0
      SSDATA(14,ISYS)= 5.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 0.5
      SSDATA(18,ISYS)= 11.04
      SSDATA(27,ISYS)= 0.305
      SSDATA(28,ISYS)= 0.296
      SSDATA(29,ISYS)= 0.955
      SSDATA(30,ISYS)= 0.9
C
C  23. SOLID POLYMER-STATIC FEED
C
      ISYS=23
      FNAME(ISYS)='SOLID POL-SF'
      SSDATA( 1,ISYS)= 1250.
      SSDATA( 2,ISYS)= 1250.
      SSDATA( 3,ISYS)= 240.
      SSDATA( 4,ISYS)= 82.0
      SSDATA( 5,ISYS)= 1.5
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 4.0
      SSDATA(11,ISYS)= 4.0
      SSDATA(12,ISYS)= 6.0
      SSDATA(13,ISYS)= 6.0
      SSDATA(14,ISYS)= 6.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 0.3
      SSDATA(18,ISYS)= 11.04
      SSDATA(27,ISYS)= 0.454
      SSDATA(28,ISYS)= 0.609
      SSDATA(29,ISYS)= 0.964
      SSDATA(30,ISYS)= 0.9
C
C  24. O2 STORAGE
C
      ISYS=24
      FNAME(ISYS)='O2 STORAGE'
      SSDATA( 1,ISYS)= 0.0
      SSDATA( 2,ISYS)= 0.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 0.0
      SSDATA( 5,ISYS)= 0.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
```

```
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(18,ISYS)= 11.04
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.0
C
C  CO2 REMOVAL
C
C    1. EDC
C
      ISYS= 1
      FNAME(ISYS)='EDC'
      SSDATA( 1,ISYS)= 80.
      SSDATA( 2,ISYS)= 80.
      SSDATA( 3,ISYS)= 360.
      SSDATA( 4,ISYS)= 115.0
      SSDATA( 5,ISYS)= 3.1
      SSDATA( 6,ISYS)= 3.8
      SSDATA( 7,ISYS)= 0.23
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 5.0
      SSDATA(11,ISYS)= 6.0
      SSDATA(12,ISYS)= 6.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 2.40
      SSDATA(19,ISYS)= 13.2
C 0.0 = USE STOICHIOMETRIC VALUE (58.5)
      SSDATA(21,ISYS)= 0.0
C 0.0 = USE STOICHIOMETRIC VALUE (540.)
      SSDATA(22,ISYS)= 0.0
C 0.0 = USE STOICHIOMETRIC VALUE (540.)
      SSDATA(23,ISYS)= 0.0
      SSDATA(27,ISYS)= 0.643
      SSDATA(28,ISYS)= 0.516
      SSDATA(29,ISYS)= 0.500
      SSDATA(30,ISYS)= 1.0
C
C    2. SAWD
C
      ISYS= 2
      FNAME(ISYS)='SAWD'
      SSDATA( 1,ISYS)= 780.
      SSDATA( 2,ISYS)= 780.
      SSDATA( 3,ISYS)= 869.
      SSDATA( 4,ISYS)= 165.0
      SSDATA( 5,ISYS)= 13.3
      SSDATA( 6,ISYS)= 3.8
      SSDATA( 7,ISYS)= 0.15
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 4.0
      SSDATA(11,ISYS)= 4.0
```

```
      SSDATA(12,ISYS)= 4.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 1.0
      SSDATA(16,ISYS)= 1.00
      SSDATA(19,ISYS)= 13.2
      SSDATA(23,ISYS)= 5184.0
      SSDATA(27,ISYS)= 0.667
      SSDATA(28,ISYS)= 0.722
      SSDATA(29,ISYS)= 0.718
      SSDATA(30,ISYS)= 1.0
C
C    3. HYDROPHOBIC MOLE SIEVE
C
      ISYS= 3
      FNAME(ISYS)='HYDRO MOL SV'
      SSDATA( 1,ISYS)= 200.
      SSDATA( 2,ISYS)= 200.
      SSDATA( 3,ISYS)= 200.
      SSDATA( 4,ISYS)= 155.0
      SSDATA( 5,ISYS)= 9.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 1.0
      SSDATA(11,ISYS)= 1.0
      SSDATA(12,ISYS)= 2.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 1.0
      SSDATA(19,ISYS)= 13.2
      SSDATA(23,ISYS)= 0.0
      SSDATA(27,ISYS)= 0.710
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 0.600
      SSDATA(30,ISYS)= 1.0
C
C    4. LIOH
C
      ISYS= 4
      FNAME(ISYS)='LIOH'
      SSDATA( 1,ISYS)= 1.
      SSDATA( 2,ISYS)= 1.
      SSDATA( 3,ISYS)= 210.
      SSDATA( 4,ISYS)= 1784.0
      SSDATA( 5,ISYS)= 75.2
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 1744.4
      SSDATA( 9,ISYS)= 69.1
      SSDATA(10,ISYS)= 8.0
      SSDATA(11,ISYS)= 8.0
      SSDATA(12,ISYS)= 8.0
      SSDATA(13,ISYS)= 8.0
      SSDATA(14,ISYS)= 6.0
      SSDATA(15,ISYS)= 1.0
      SSDATA(16,ISYS)= 60.0
      SSDATA(19,ISYS)= 13.2
```

```
      SSDATA(23,ISYS)= 465.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C     5. FOUR BED MOLE SIEVE
C
      ISYS= 5
      FNAME(ISYS)='4 BED MOL SV'
      SSDATA( 1,ISYS)= 0.0
      SSDATA( 2,ISYS)= 0.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 0.0
      SSDATA( 5,ISYS)= 0.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(19,ISYS)= 13.2
      SSDATA(23,ISYS)= 0.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C     6. SKYLAB MOLE SIEVE
C
      ISYS= 6
      FNAME(ISYS)='SKLAB MOL SV'
      SSDATA( 1,ISYS)= 0.0
      SSDATA( 2,ISYS)= 0.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 0.0
      SSDATA( 5,ISYS)= 0.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(19,ISYS)= 13.2
      SSDATA(23,ISYS)= 0.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
```

```
C  CO2 REDUCTION
C
C  11. BOSCH
C
      ISYS=11
      FNAME(ISYS)='BOSCH'
      SSDATA( 1,ISYS)= 140.
      SSDATA( 2,ISYS)= 140.
      SSDATA( 3,ISYS)= 170.
      SSDATA( 4,ISYS)= 446.0
      SSDATA( 5,ISYS)= 25.5
      SSDATA( 6,ISYS)= 108.0
      SSDATA( 7,ISYS)= 4.71
      SSDATA( 8,ISYS)= 20.0
      SSDATA( 9,ISYS)= 12.0
      SSDATA(10,ISYS)= 4.0
      SSDATA(11,ISYS)= 4.0
      SSDATA(12,ISYS)= 4.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 1.0
      SSDATA(16,ISYS)= 20.2
      SSDATA(19,ISYS)= 13.2
C 0.0 = USE STOICHIOMETRIC VALUE (116.1)
      SSDATA(21,ISYS)= 0.0
C 0.0 = USE STOICHIOMETRIC VALUE (1026.)
      SSDATA(23,ISYS)= 0.0
C 0.0 = USE STOICHIOMETRIC VALUE (331.2)
      SSDATA(24,ISYS)= 0.0
      SSDATA(26,ISYS)= 0.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C  12. SABATIER
C
      ISYS=12
      FNAME(ISYS)='SABATIER'
      SSDATA( 1,ISYS)= 90.
      SSDATA( 2,ISYS)= 90.
      SSDATA( 3,ISYS)= 97.5
      SSDATA( 4,ISYS)= 256.75
      SSDATA( 5,ISYS)= 19.27
      SSDATA( 6,ISYS)= 64.0
      SSDATA( 7,ISYS)= 8.35
      SSDATA( 8,ISYS)= 5.75
      SSDATA( 9,ISYS)= 2.4
      SSDATA(10,ISYS)= 5.0
      SSDATA(11,ISYS)= 5.0
      SSDATA(12,ISYS)= 4.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 1.0
      SSDATA(16,ISYS)= 11.2
      SSDATA(19,ISYS)= 13.2
C 0.0 = USE STOICHIOMETRIC VALUE (158.4)
      SSDATA(21,ISYS)= 0.0
C 0.0 = USE STOICHIOMETRIC VALUE (699.3)
      SSDATA(23,ISYS)= 0.0
```

```
C 0 0 = USE STOICHIOMETRIC VALUE (316.8)
      SSDATA(25,ISYS)= 0.0
      SSDATA(26,ISYS)= 356.85
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C  13. SABATIER/CH4 DISSOCIATION
C
      ISYS=13
      FNAME(ISYS)='SABAT/CH4 DS'
      SSDATA( 1,ISYS)= 0.0
      SSDATA( 2,ISYS)= 0.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 0.0
      SSDATA( 5,ISYS)= 0.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(19,ISYS)= 13.5
      SSDATA(21,ISYS)= 0.0
      SSDATA(23,ISYS)= 0.0
      SSDATA(24,ISYS)= 0.0
      SSDATA(25,ISYS)= 0.0
      SSDATA(26,ISYS)= 0.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.0
C
C  WATER RECLAMATION-PHASE CHANGE
C
C  31. VCD
C
      ISYS=31
      FNAME(ISYS)='VCD'
      SSDATA( 1,ISYS)= 214.
      SSDATA( 2,ISYS)= 214.
      SSDATA( 3,ISYS)= 214.
      SSDATA( 4,ISYS)= 327.0
      SSDATA( 5,ISYS)= 25.6
      SSDATA( 6,ISYS)= 71.3
      SSDATA( 7,ISYS)= 1.3
      SSDATA( 8,ISYS)= 111.6
      SSDATA( 9,ISYS)= 1.8
      SSDATA(10,ISYS)= 2.0
      SSDATA(11,ISYS)= 2.0
      SSDATA(12,ISYS)= 2.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 2.0
      SSDATA(15,ISYS)= 4.0
```

```
      SSDATA(16,ISYS)= 30.0
      SSDATA(20,ISYS)= 26.4
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.96
C
C  32. TIMES
C
      ISYS=32
      FNAME(ISYS)='TIMES'
      SSDATA( 1,ISYS)= 351.
      SSDATA( 2,ISYS)= 351.
      SSDATA( 3,ISYS)= 351.
      SSDATA( 4,ISYS)= 359.0
      SSDATA( 5,ISYS)= 20.0
      SSDATA( 6,ISYS)= 159.14
      SSDATA( 7,ISYS)= 2.6
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 4.0
      SSDATA(11,ISYS)= 4.0
      SSDATA(12,ISYS)= 4.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 4.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 20.0
      SSDATA(20,ISYS)= 26.4
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.939
C
C  33. AIR EVAP
C
      ISYS=33
      FNAME(ISYS)='AIR EVAP'
      SSDATA( 1,ISYS)= 2260.
      SSDATA( 2,ISYS)= 2260.
      SSDATA( 3,ISYS)= 2260.
      SSDATA( 4,ISYS)= 221.0
      SSDATA( 5,ISYS)= 15.0
      SSDATA( 6,ISYS)= 124.6
      SSDATA( 7,ISYS)= 2.2
      SSDATA( 8,ISYS)= 16.6
      SSDATA( 9,ISYS)= 0.28
      SSDATA(10,ISYS)= 6.0
      SSDATA(11,ISYS)= 4.0
      SSDATA(12,ISYS)= 6.0
      SSDATA(13,ISYS)= 4.0
      SSDATA(14,ISYS)= 6.0
      SSDATA(15,ISYS)= 4.0
      SSDATA(16,ISYS)= 10.0
      SSDATA(20,ISYS)= 26.4
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C  34. VPCAR
```

```
C
      ISYS=34
      FNAME(ISYS)='VPCAR'
      SSDATA( 1,ISYS)= 0.0
      SSDATA( 2,ISYS)= 0.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 0.0
      SSDATA( 5,ISYS)= 0.0
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(20,ISYS)= 26.4
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.0
C
C  WATER RECLAMATION-NO PHASE CHANGE
C
C  41. MULTIFILTRATION
C
      ISYS=41
      FNAME(ISYS)='MULTIFILTRTN'
      SSDATA( 1,ISYS)= 30.
      SSDATA( 2,ISYS)= 30.0
      SSDATA( 3,ISYS)= 30.
      SSDATA( 4,ISYS)= 98.7
      SSDATA( 5,ISYS)= 2.7
      SSDATA( 6,ISYS)= 9.5
      SSDATA( 7,ISYS)= 0.17
      SSDATA( 8,ISYS)= 10.9
      SSDATA( 9,ISYS)= 0.23
      SSDATA(10,ISYS)= 7.0
      SSDATA(11,ISYS)= 7.0
      SSDATA(12,ISYS)= 7.0
      SSDATA(13,ISYS)= 7.0
      SSDATA(14,ISYS)= 7.0
      SSDATA(15,ISYS)= 7.0
      SSDATA(16,ISYS)= 8.00
      SSDATA(20,ISYS)= 72.0
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 1.0
C
C  42. HYPERFILTRATION
C
      ISYS=42
      FNAME(ISYS)='HYPERFILTRTN'
      SSDATA( 1,ISYS)= 272.0
      SSDATA( 2,ISYS)= 272.0
      SSDATA( 3,ISYS)= 0.0
```

```
      SSDATA( 4,ISYS)= 460.0
      SSDATA( 5,ISYS)= 30.8
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(20,ISYS)= 274.02
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.92
C
C  43. REVERSE OSMOSIS
C
      ISYS=43
      FNAME(ISYS)='REV OSMOSIS'
      SSDATA( 1,ISYS)= 45.0
      SSDATA( 2,ISYS)= 45.0
      SSDATA( 3,ISYS)= 0.0
      SSDATA( 4,ISYS)= 125.0
      SSDATA( 5,ISYS)= 30.8
      SSDATA( 6,ISYS)= 0.0
      SSDATA( 7,ISYS)= 0.0
      SSDATA( 8,ISYS)= 0.0
      SSDATA( 9,ISYS)= 0.0
      SSDATA(10,ISYS)= 0.0
      SSDATA(11,ISYS)= 0.0
      SSDATA(12,ISYS)= 0.0
      SSDATA(13,ISYS)= 0.0
      SSDATA(14,ISYS)= 0.0
      SSDATA(15,ISYS)= 0.0
      SSDATA(16,ISYS)= 0.0
      SSDATA(20,ISYS)= 274.02
      SSDATA(27,ISYS)= 1.0
      SSDATA(28,ISYS)= 1.0
      SSDATA(29,ISYS)= 1.0
      SSDATA(30,ISYS)= 0.96
C
      RETURN
      END
C
      SUBROUTINE AUXILS
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
```

```
     * GWATER(NAAUX,NAAUX),PEWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
       CHARACTER*6 ADUMP
C
C  WATER RECLAMATION SYSTEM COMPONENT DATA
       WRSAUX(1,0)=0.0
C TANKAGE WEIGHT (LB/LB H2O)
       WRSAUX( 1,1)= 0.2833
C TANKAGE VOLUME (FT3/LB H2O)
       WRSAUX( 1,2)= 0.0160
C TANKAGE POWER (WATTS/LB H2O)
       WRSAUX( 1,3)= 0.0
C
       WRSAUX(2,0)=0.0
C PRIMARY PUMP WEIGHT (LB)
       WRSAUX( 2,1)= 5.0
C PRIMARY PUMP VOLUME (FT3)
       WRSAUX( 2,2)= 0.030
C PRIMARY PUMP POWER (WATTS)
       WRSAUX( 2,3)= 15.0
C
       WRSAUX(3,0)=0.0
C SECONDARY PUMP WEIGHT (LB)
       WRSAUX( 3,1)= 3.0
C SECONDARY PUMP VOLUME (FT3)
       WRSAUX( 3,2)= 0.0368
C SECONDARY PUMP POWER (WATTS)
       WRSAUX( 3,3)= 28.0
C
       WRSAUX(4,0)=0.0
C RECIRCULATION PUMP WEIGHT (LB)
       WRSAUX( 4,1)= 1.4
C RECIRCULATION PUMP VOLUME (FT3)
       WRSAUX( 4,2)= 0.0368
C RECIRCULATION PUMP POWER (WATTS)
       WRSAUX( 4,3)= 10.0
C
       WRSAUX(5,0)=0.0
C AIR-H2O SEPARATOR WEIGHT (LB)
       WRSAUX( 5,1)= 17.3
C AIR-H2O SEPARATOR VOLUME (FT3)
       WRSAUX( 5,2)= 0.8542
C AIR-H2O SEPARATOR POWER (WATTS)
       WRSAUX( 5,3)= 47.0
C
       WRSAUX(6,0)=0.0
C BIOCIDE RESERVOIR WEIGHT (LB)
       WRSAUX( 6,1)= 5.0
C BIOCIDE RESERVOIR VOLUME (FT3)
       WRSAUX( 6,2)= 0.2471
C BIOCIDE RESERVOIR POWER (WATTS)
       WRSAUX( 6,3)= 0.0
C
       WRSAUX(7,0)=0.0
C PRETREAT RESERVOIR WEIGHT (LB)
       WRSAUX( 7,1)= 5.0
C PRETREAT RESERVOIR VOLUME (FT3)
       WRSAUX( 7,2)= 0.2471
C PRETREAT RESERVOIR POWER (WATTS)
       WRSAUX( 7,3)= 0.0
C
```

```
      WRSAUX(6,0)=0.0
C SPRING LOADED ACCUMULATOR WEIGHT (LB)
      WRSAUX( 8,1)= 0.5
C SPRING LOADED ACCUMULATOR VOLUME (FT3)
      WRSAUX( 8,2)= 0.0250
C SPRING LOADED ACCUMULATOR POWER (WATTS)
      WRSAUX( 8,3)= 0.0
C
      WRSAUX(9,0)=0.0
C SHUT OFF VALVE WEIGHT (LB)
      WRSAUX( 9,1)= 1.42
C SHUT OFF VALVE VOLUME (FT3)
      WRSAUX( 9,2)= 0.00796
C SHUT OFF VALVE POWER (WATTS)
      WRSAUX( 9,3)= 0.0
C
      WRSAUX(10,0)=0.0
C SOLENOID VALVE WEIGHT (LB)
      WRSAUX(10,1)= 1.42
C SOLENOID VALVE VOLUME (FT3)
      WRSAUX(10,2)= 0.00796
C SOLENOID VALVE POWER (WATTS)-ASSUMES 5% DUTY
      WRSAUX(10,3)= 2.50
C
      WRSAUX(11,0)=0.0
C CHECK VALVE WEIGHT (LB)
      WRSAUX(11,1)= 0.14
C CHECK VALVE VOLUME (FT3)
      WRSAUX(11,2)= 0.00110
C CHECK VALVE POWER (WATTS)
      WRSAUX(11,3)= 0.0
C
      WRSAUX(12,0)=0.0
C PREFILTER PACKAGE WEIGHT (LB)
      WRSAUX(12,1)= 9.5
C PREFILTER PACKAGE VOLUME (FT3)
      WRSAUX(12,2)= 0.1700
C PREFILTER PACKAGE POWER (WATTS)
      WRSAUX(12,3)= 0.0
C
      WRSAUX(13,0)=0.0
C CARBON & ION EXCHANGE WEIGHT (LB)
      WRSAUX(13,1)= 10.0
C CARBON & ION EXCHANGE VOLUME (FT3)
      WRSAUX(13,2)= 0.2300
C CARBON & ION EXCHANGE POWER (WATTS)
      WRSAUX(13,3)= 0.0
C
      WRSAUX(14,0)=0.0
C I2 DISPENSE/DETECT WEIGHT (LB)
      WRSAUX(14,1)= 2.7
C I2 DISPENSE/DETECT VOLUME (FT3)
      WRSAUX(14,2)= 0.08
C I2 DISPENSE/DETECT POWER (WATTS)
      WRSAUX(14,3)= 5.0
C
      WRSAUX(15,0)=0.0
C SAMPLE PORT WEIGHT (LB)
      WRSAUX(15,1)= 0.1
C SAMPLE PORT VOLUME (FT3)
```

```
      WRSAUX(15,2)= 0.00401
C SAMPLE PORT POWER (WATTS)
      WRSAUX(15,3)= 0.0
C
      WRSAUX(16,0)=0.0
C LINE/DUCT WEIGHT (LB/FT) - ASSUMES .5"D
      WRSAUX(16,1)= 0.42
C LINE/DUCT VOLUME (FT3/FT)- ASSUMES .5"D
      WRSAUX(16,2)= 0.00249
C LINE/DUCT POWER (WATTS/FT)-ASSUMES .5"D
      WRSAUX(16,3)= 0.0
C
C    IWAT = NUMBER DENOTING TYPE OF WATER BEING RECOVERED
C  WRSAUXQ(IWAT, 0)= NUMBER OF RECYCLES H2O/RESUPPLY INTERVAL
C  WRSAUXQ(IWAT, 1)= NUMBER OF WATER TANKS
C  WRSAUXQ(IWAT, 2)= NUMBER OF PRIMARY PUMPS
C  WRSAUXQ(IWAT, 3)= NUMBER OF SECONDARY PUMPS
C  WRSAUXQ(IWAT, 4)= NUMBER OF RECIRCULATION PUMPS
C  WRSAUXQ(IWAT, 5)= NUMBER OF AIR-H2O SEPARATORS
C  WRSAUXQ(IWAT, 6)= NUMBER OF BIOCIDE RESERVOIRS
C  WRSAUXQ(IWAT, 7)= NUMBER OF PRETREAT RESERVOIRS
C  WRSAUXQ(IWAT, 8)= NUMBER OF SPRING LOADED ACCUMULATORS
C  WRSAUXQ(IWAT, 9)= NUMBER OF SHUT OFF VALVES
C  WRSAUXQ(IWAT,10)= NUMBER OF SOLENOID VALVES
C  WRSAUXQ(IWAT,11)= NUMBER OF CHECK VALVES
C  WRSAUXQ(IWAT,12)= NUMBER OF PREFILTER PACKAGES
C  WRSAUXQ(IWAT,13)= NUMBER OF CARBON & ION EXCHANGERS
C  WRSAUXQ(IWAT,14)=  UMBER OF I2 DISPENSERS/DETECTORS
C  WRSAUXQ(IWAT,15)=  UMBER OF SAMPLE PORTS
C  WRSAUXQ(IWAT,16)= LENGTH OF LINE/DUCT (FT)
C
C   1. CONDENSATE QUANTITIES OF AUXILIARY EQUIPMENT
      IWAT=1
      WRSAUXQ(IWAT, 0)= 89.
      WRSAUXQ(IWAT, 1)= 2.
      WRSAUXQ(IWAT, 2)= 2.
      WRSAUXQ(IWAT, 3)= 2.
      WRSAUXQ(IWAT, 4)= 2.
      WRSAUXQ(IWAT, 5)= 2.
      WRSAUXQ(IWAT, 6)= 2.
      WRSAUXQ(IWAT, 7)= 2.
      WRSAUXQ(IWAT, 8)= 4.
      WRSAUXQ(IWAT, 9)= 6.
      WRSAUXQ(IWAT,10)= 30.
      WRSAUXQ(IWAT,11)= 2.
      WRSAUXQ(IWAT,12)= 2.
C ASSUMED PART OF WATER REC SYSTEM
      WRSAUXQ(IWAT,13)= 0.
      WRSAUXQ(IWAT,14)= 2.
      WRSAUXQ(IWAT,15)= 2.
      WRSAUXQ(IWAT,16)= 56.
C
C   2. REDUCTION H2O QUANTITIES OF AUXILIARY EQUIPMENT
C       (TOTALS INCLUDED WITH AIR REVITALIZATION SYSTEM)
      IWAT=2
      WRSAUXQ(IWAT, 0)= 0.0
      WRSAUXQ(IWAT, 1)= 0.0
      WRSAUXQ(IWAT, 2)= 0.0
      WRSAUXQ(IWAT, 3)= 0.0
      WRSAUXQ(IWAT, 4)= 0.0
```

```
      WRSAUXQ(IWAT, 5)= 0.0
      WRSAUXQ(IWAT, 6)= 0.0
      WRSAUXQ(IWAT, 7)= 0.0
      WRSAUXQ(IWAT, 8)= 0.0
      WRSAUXQ(IWAT, 9)= 0.0
      WRSAUXQ(IWAT,10)= 0.0
      WRSAUXQ(IWAT,11)= 0.0
      WRSAUXQ(IWAT,12)= 0.0
C ASSUMED PART OF WATER REC SYSTEM
      WRSAUXQ(IWAT,13)= 0.0
      WRSAUXQ(IWAT,14)= 0.0
      WRSAUXQ(IWAT,15)= 0.0
      WRSAUXQ(IWAT,16)= 0.0
C
C   3. HYGIENE H2O QUANTITIES OF AUXILIARY EQUIPMENT
      IWAT=3
      WRSAUXQ(IWAT, 0)= 89.
      WRSAUXQ(IWAT, 1)= 2.
      WRSAUXQ(IWAT, 2)= 2.
      WRSAUXQ(IWAT, 3)= 2.
      WRSAUXQ(IWAT, 4)= 2.
      WRSAUXQ(IWAT, 5)= 5.
      WRSAUXQ(IWAT, 6)= 2.
      WRSAUXQ(IWAT, 7)= 2.
      WRSAUXQ(IWAT, 8)= 4.
      WRSAUXQ(IWAT, 9)= 26.
      WRSAUXQ(IWAT,10)= 28.
      WRSAUXQ(IWAT,11)= 5.
      WRSAUXQ(IWAT,12)= 5.
C ASSUMED PART OF WATER REC SYSTEM
      WRSAUXQ(IWAT,13)= 0.
      WRSAUXQ(IWAT,14)= 2.
      WRSAUXQ(IWAT,15)= 2.
      WRSAUXQ(IWAT,16)= 108.
C
C   4. CLOTHES WASH H2O QUANTITIES OF AUXILIARY EQUIPMENT
      IWAT=4
      WRSAUXQ(IWAT, 0)= 89.
      WRSAUXQ(IWAT, 1)= 2.
      WRSAUXQ(IWAT, 2)= 1.
      WRSAUXQ(IWAT, 3)= 0.
      WRSAUXQ(IWAT, 4)= 1.
      WRSAUXQ(IWAT, 5)= 1.
      WRSAUXQ(IWAT, 6)= 1.
      WRSAUXQ(IWAT, 7)= 1.
      WRSAUXQ(IWAT, 8)= 2.
      WRSAUXQ(IWAT, 9)= 5.
      WRSAUXQ(IWAT,10)= 7.
      WRSAUXQ(IWAT,11)= 1.
      WRSAUXQ(IWAT,12)= 1.
C C SUMED PART OF WATER REC SYSTEM
      WRSAUXQ(IWAT,13)= 0.
      WRSAUXQ(IWAT,14)= 1.
      WRSAUXQ(IWAT,15)= 1.
      WRSAUXQ(IWAT,16)= 24.
C
C   5. URINE H2O QUANTITIES OF AUXILIARY EQUIPMENT
      IWAT=5
      WRSAUXQ(IWAT, 0)= 89.
      WRSAUXQ(IWAT, 1)= 2.
```

```
      WRSAUXQ(IWAT, 2)= 0.
      WRSAUXQ(IWAT, 3)= 2.
      WRSAUXQ(IWAT, 4)= 4.
      WRSAUXQ(IWAT, 5)= 6.
      WRSAUXQ(IWAT, 6)= 2.
      WRSAUXQ(IWAT, 7)= 2.
      WRSAUXQ(IWAT, 8)= 4.
      WRSAUXQ(IWAT, 9)= 28.
      WRSAUXQ(IWAT,10)= 20.
      WRSAUXQ(IWAT,11)= 6.
      WRSAUXQ(IWAT,12)= 8.
      WRSAUXQ(IWAT,13)= 8.
      WRSAUXQ(IWAT,14)= 4.
      WRSAUXQ(IWAT,15)= 4.
      WRSAUXQ(IWAT,16)= 98.
C
C    6. DISHWASH H20 QUANTITIES OF AUXILIARY EQUIPMENT
      IWAT=6
      WRSAUXQ(IWAT, 0)= 89.
      WRSAUXQ(IWAT, 1)= 2.
      WRSAUXQ(IWAT, 2)= 1.
      WRSAUXQ(IWAT, 3)= 0.
      WRSAUXQ(IWAT, 4)= 1.
      WRSAUXQ(IWAT, 5)= 1.
      WRSAUXQ(IWAT, 6)= 1.
      WRSAUXQ(IWAT, 7)= 1.
      WRSAUXQ(IWAT, 8)= 2.
      WRSAUXQ(IWAT, 9)= 5.
      WRSAUXQ(IWAT,10)= 7.
      WRSAUXQ(IWAT,11)= 1.
      WRSAUXQ(IWAT,12)= 1.
C ASSUMED PART OF WATER REC SYSTEM
      WRSAUXQ(IWAT,13)= 0.
      WRSAUXQ(IWAT,14)= 1.
      WRSAUXQ(IWAT,15)= 1.
      WRSAUXQ(IWAT,16)= 24.
C
C
C
C  AIR REVITALIZATION SYSTEM COMPONENT DATA
C WATER PUMP WEIGHT (LB)
      ARSAUX( 1,1)= 3.0
C WATER PUMP VOLUME (FT3)
      ARSAUX( 1,2)= 0.04
C WATER PUMP POWER (WATTS)
      ARSAUX( 1,3)= 28.
C HOT WATER H/X WEIGHT (LB)
      ARSAUX( 2,1)= 26.2
C HOT WATER H/X VOLUME (FT3)
      ARSAUX( 2,2)= 0.25
C HOT WATER H/X POWER (WATTS)
      ARSAUX( 2,3)= 0.0
C PURIFICATION DEVICE WEIGHT (LB)
      ARSAUX( 3,1)= 1.2
C PURIFICATION DEVICE VOLUME (FT3)
      ARSAUX( 3,2)= 0.10
C PURIFICATION DEVICE POWER (WATTS)
      ARSAUX( 3,3)= 0.0
C O2 COMPRESSOR WEIGHT (LB)
      ARSAUX( 4,1)= 28.0
```

```
C O2 COMPRESSOR VOLUME (FT3)
      ARSAUX( 4,2)= 0.8
C O2 COMPRESSOR POWER (WATTS)-MAX DUTY CYCLE
      ARSAUX( 4,3)= 302.
C O2 STORAGE TANK WEIGHT (LB)
      ARSAUX( 5,1)= 39.8
C O2 STORAGE TANK VOLUME (FT3)
      ARSAUX( 5,2)= 1.4
C O2 STORAGE TANK POWER (WATTS)
      ARSAUX( 5,3)= 0.0
C H2 COMPRESSOR WEIGHT (LB)
      ARSAUX( 6,1)= 28.0
C H2 COMPRESSOR VOLUME (FT3)
      ARSAUX( 6,2)= 0.8
C H2 COMPRESSOR POWER (WATTS)-MAX DUTY CYCLE
      ARSAUX( 6,3)= 302.
C H2 STORAGE TANK WEIGHT (LB)
      ARSAUX( 7,1)= 124.
C H2 STORAGE TANK VOLUME (FT3)
      ARSAUX( 7,2)= 5.5
C H2 STORAGE TANK POWER (WATTS)
      ARSAUX( 7,3)= 0.0
C CH4/CO2 COMPRESSOR WEIGHT (LB)
      ARSAUX( 8,1)= 0.0
C CH4/CO2 COMPRESSOR VOLUME (FT3)
      ARSAUX( 8,2)= 0.0
C CH4/CO2 COMPRESSOR POWER (WATTS)-MAX DUTY
      ARSAUX( 8,3)= 0.0
C CH4/CO2 STORAGE TANK WEIGHT (LB)
      ARSAUX( 9,1)= 0.0
C CH4/CO2 STORAGE TANK VOLUME (FT3)
      ARSAUX( 9,2)= 0.0
C CH4/CO2 STORAGE TANK POWER (WATTS)
      ARSAUX( 9,3)= 0.0
C CARBON STORAGE EQUIPMENT WEIGHT (LB)
      ARSAUX(10,1)= 0.0
C CARBON STORAGE EQUIPMENT VOLUME (FT3)
      ARSAUX(10,2)= 0.0
C CARBON STORAGE EQUIPMENT POWER (WATTS)
      ARSAUX(10,3)= 0.0
C
C     IAIR = NUMBER DENOTING TYPE OF WATER BEING PROCESSED
C
C  ARSAUXQ(IAIR, 1)= NUMBER OF WATER PUMPS
C  ARSAUXQ(IAIR, 2)= NUMBER OF HOT WATER HEAT EXCHANGERS
C  ARSAUXQ(IAIR, 3)= NUMBER OF PURIFICATION DEVICES
C  ARSAUXQ(IAIR, 4)= NUMBER OF O2 COMPRESSORS
C  ARSAUXQ(IAIR, 5)= NUMBER OF O2 STORAGE TANKS
C  ARSAUXQ(IAIR, 6)= NUMBER OF H2 COMPRESSORS
C  ARSAUXQ(IAIR, 7)= NUMBER OF H2 STORAGE TANKS
C  ARSAUXQ(IAIR, 8)= NUMBER OF CH4/CO2 COMPRESSORS
C  ARSAUXQ(IAIR, 9)= NUMBER OF CH4/CO2 STORAGE TANKS
C  ARSAUXQ(IAIR,10)= NUMBER OF CARBON STORAGE EQUIPMENTS
C
C     11. BOSCH
      IAIR=11
      ARSAUXQ(IAIR, 1)=2.
      ARSAUXQ(IAIR, 6)=2.
      ARSAUXQ(IAIR, 7)=2.
      ARSAUXQ(IAIR,10)=2.
```

```
C
C     12. SABATIER
      IAIR=12
      ARSAUXQ(IAIR, 1)=2.
      ARSAUXQ(IAIR, 8)=2.
      ARSAUXQ(IAIR, 9)=2.
C
C     13. SABATIER/CH4 DISSOCIATION
      IAIR=13
      ARSAUXQ(IAIR, 1)=2.
      ARSAUXQ(IAIR, 8)=2.
      ARSAUXQ(IAIR, 9)=2.
      ARSAUXQ(IAIR,10)=2.
C
C     21. STATIC FEED-WES
      IAIR=21
      ARSAUXQ(IAIR, 2)=2.
      ARSAUXQ(IAIR, 3)=2.
      ARSAUXQ(IAIR, 4)=2.
      ARSAUXQ(IAIR, 5)=2.
      ARSAUXQ(IAIR, 6)=2.
      ARSAUXQ(IAIR, 7)=2.
C
C     22. SOLID POLYMER - LIQUID FEED
      IAIR=22
      ARSAUXQ(IAIR, 2)=2.
      ARSAUXQ(IAIR, 3)=2.
      ARSAUXQ(IAIR, 4)=2.
      ARSAUXQ(IAIR, 5)=2.
      ARSAUXQ(IAIR, 6)=2.
      ARSAUXQ(IAIR, 7)=2.
C
C     23. SOLID POLYMER - STATIC FEED
      IAIR=23
      ARSAUXQ(IAIR, 2)=2.
      ARSAUXQ(IAIR, 3)=2.
      ARSAUXQ(IAIR, 4)=2.
      ARSAUXQ(IAIR, 5)=2.
      ARSAUXQ(IAIR, 6)=2.
      ARSAUXQ(IAIR, 7)=2.
      RETURN
      END
C
      SUBROUTINE SBRSGN(NWRITE)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER(NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
      NREAD=5
    5 WRITE(NWRITE,10)
   10 FORMAT(////,T20,'ECLSS AVERAGE DESIGN LOADS')
```

```
      K=1
      WRITE(NWRITE,100) K,DESIGN(K)
  100 FORMAT(T2,I2,T5,'N, NUMBER OF CREW                                ',
     * T55,F8.2)
      K=2
      WRITE(NWRITE,110) K,DESIGN(K)
  110 FORMAT(T2,I2,T5,'R, RESUPPLY INTERVAL(DAYS)                       ',
     * T55,F8.2)
      K=3
      WRITE(NWRITE,120) K,DESIGN(K)
  120 FORMAT(T2,I2,T5,'M, MISSION DURATION(YEARS)                       ',
     * T55,F8.2)
      K=4
      WRITE(NWRITE,130) K,DESIGN(K)
  130 FORMAT(T2,I2,T5,'N2 CABIN PARTIAL PRESSURE(PSIA)                  ',
     * T55,F8.2)
      K=5
      WRITE(NWRITE,140) K,DESIGN(K)
  140 FORMAT(T2,I2,T5,'O2 CABIN PARTIAL PRESSURE(PSIA)                  ',
     * T55,F8.2)
      K=6
      WRITE(NWRITE,150) K,DESIGN(K)
  150 FORMAT(T2,I2,T5,'CABIN LEAKAGE(LB/DAY)                            ',
     * T55,F8.2)
      K=7
      WRITE(NWRITE,160) K,DESIGN(K)
  160 FORMAT(T2,I2,T5,'AIR LOCK GAS LOSS(LB/DAY)                        ',
     * T55,F8.2)
      K=8
      WRITE(NWRITE,170) K,DESIGN(K)
  170 FORMAT(T2,I2,T5,'METABOLIC O2(LB/MAN-DAY)                         ',
     * T55,F8.2)
      K=9
      WRITE(NWRITE,180) K,DESIGN(K)
  180 FORMAT(T2,I2,T5,'METABOLIC CO2(LB/MAN-DAY)                        ',
     * T55,F8.2)
      K=10
      WRITE(NWRITE,190) K,DESIGN(K)
  190 FORMAT(T2,I2,T5,'RE, EVA MAN-HOURS/RESUPPLY INTERVAL              ',
     * T55,F8.2)
      K=11
      WRITE(NWRITE,200) K,DESIGN(K)
  200 FORMAT(T2,I2,T5,'EVA O2(LB/MAN-HOUR)                              ',
     * T55,F8.2)
      K=12
      WRITE(NWRITE,210) K,DESIGN(K)
  210 FORMAT(T2,I2,T5,'EVA CO2(LB/MAN-HOUR)                             ',
     * T55,F8.2)
      K=13
      WRITE(NWRITE,220) K,DESIGN(K)
  220 FORMAT(T2,I2,T5,'EVA H2O(LB/MAN-HOUR)                             ',
     * T55,F8.2)
      K=14
      WRITE(NWRITE,230) K,DESIGN(K)
  230 FORMAT(T2,I2,T5,'EVA WATER LOST(LB/MAN-HOUR)                      ',
     * T55,F8.2)
      K=15
      WRITE(NWRITE,240) K,DESIGN(K)
  240 FORMAT(T2,I2,T5,'DRINKING WATER(LB/MAN-DAY)                       ',
     * T55,F8.2)
```

```
      K=16
      WRITE(NWRITE,250) K,DESIGN(K)
  250 FORMAT(T2,I2,T5,'FOOD PREP H2O(LB/MAN-DAY)                          ',
     * T55,F8.2)
      K=17
      WRITE(NWRITE,260) K,DESIGN(K)
  260 FORMAT(T2,I2,T5,'H2O IN FOOD(LB/MAN-DAY)                            ',
     * T55,F8.2)
      IF(NWRITE.EQ.6) CALL DSGNIN(*5)
  275 WRITE(NWRITE,2)
    2 FORMAT(4(/))
      K=18
      WRITE(NWRITE,270) K,DESIGN(K)
  270 FORMAT(T2,I2,T5,'METABOLIC H2O PRODUCTION(LB/MAN-DAY)               ',
     * T55,F8.2)
      K=19
      WRITE(NWRITE,280) K,DESIGN(K)
  280 FORMAT(T2,I2,T5,'HYGIENE LATENT H2O(LB/MAN-DAY)                     ',
     * T55,F8.2)
      K=20
      WRITE(NWRITE,290) K,DESIGN(K)
  290 FORMAT(T2,I2,T5,'FOOD PREP LATENT H2O(LB/MAN-DAY)                   ',
     * T55,F8.2)
      K=21
      WRITE(NWRITE,300) K,DESIGN(K)
  300 FORMAT(T2,I2,T5,'EXPERIMENTS LATENT H2O(LB/MAN-DAY)                 ',
     * T55,F8.2)
      K=22
      WRITE(NWRITE,310) K,DESIGN(K)
  310 FORMAT(T2,I2,T5,'LAUNDRY LATENT H2O(LB/MAN-DAY)                     ',
     * T55,F8.2)
      K=23
      WRITE(NWRITE,320) K,DESIGN(K)
  320 FORMAT(T2,I2,T5,'DISHWASHER LATENT H2O(LB/MAN-DAY)                  ',
     * T55,F8.2)
      K=24
      WRITE(NWRITE,330) K,DESIGN(K)
  330 FORMAT(T2,I2,T5,'SWEAT & RESP H2O(LB/MAN-DAY)                       ',
     * T55,F8.2)
      K=25
      WRITE(NWRITE,340) K,DESIGN(K)
  340 FORMAT(T2,I2,T5,'URINE H2O(LB/MAN-DAY)                              ',
     * T55,F8.2)
      K=26
      WRITE(NWRITE,350) K,DESIGN(K)
  350 FORMAT(T2,I2,T5,'URINAL FLUSH H2O(LB/MAN-DAY)                       ',
     * T55,F8.2)
      K=27
      WRITE(NWRITE,360) K,DESIGN(K)
  360 FORMAT(T2,I2,T5,'FECAL H2O(LB/MAN-DAY)                              ',
     * T55,F8.2)
      K=28
      WRITE(NWRITE,370) K,DESIGN(K)
  370 FORMAT(T2,I2,T5,'HANDWASH H2O(LB/MAN-DAY)                           ',
     * T55,F8.2)
      K=29
      WRITE(NWRITE,380) K,DESIGN(K)
  380 FORMAT(T2,I2,T5,'SHOWER H2O(LB/MAN-DAY)                             ',
     * T55,F8.2)
      K=30
```

```
      WRITE(NWRITE,390) K,DESIGN(K)
  390 FORMAT(T2,I2,T5,'CLOTHES WASH H2O(LB/MAN-DAY)                    ',
     * T55,F8.2)
      K=31
      WRITE(NWRITE,400) K,DESIGN(K)
  400 FORMAT(T2,I2,T5,'DISHWASHER H2O(LB/MAN-DAY)                      ',
     * T55,F8.2)
      K=32
      WRITE(NWRITE,410) K,DESIGN(K)
  410 FORMAT(T2,I2,T5,'EXPERIMENT H2O(LB/MAN-DAY)                      ',
     * T55,F8.2)
      K=33
      WRITE(NWRITE,420) K,DESIGN(K)
  420 FORMAT(T2,I2,T5,'EXPERIMENT H2O LOST(LB/MAN-DAY)                 ',
     * T55,F8.2)
      K=34
      WRITE(NWRITE,430) K,DESIGN(K)
  430 FORMAT(T2,I2,T5,'CLOTHING WEIGHT(LB/MAN-DAY)                     ',
     * T55,F8.2)
      IF(NWRITE.EQ.6) CALL DSGNIN(*275)
  445 WRITE(NWRITE,2)
      K=35
      WRITE(NWRITE,440) K,DESIGN(K)
  440 FORMAT(T2,I2,T5,'CLOTHING VOLUME(FT3/MAN-DAY)                    ',
     * T55,F8.2)
      K=36
      WRITE(NWRITE,450) K,DESIGN(K)
  450 FORMAT(T2,I2,T5,'RESUPPLY CLOTHING WITH CLOTHES WASH(DAYS)       ',
     * T55,F8.2)
      K=38
      WRITE(NWRITE,470) K,DESIGN(K)
  470 FORMAT(T2,I2,T5,'RESUPPLY DISHES W/O DISHWASH(LB/MAN-DAY)        ',
     * T55,F8.2)
      K=39
      WRITE(NWRITE,480) K,DESIGN(K)
  480 FORMAT(T2,I2,T5,'RESUPPLY DISHES W/O DISHWASH(LB/MAN-DAY)        ',
     * T55,F8.2)
      K=40
      WRITE(NWRITE,490) K,DESIGN(K)
  490 FORMAT(T2,I2,T5,'TRASH WEIGHT(LB/MAN-DAY)                       ',,
     * T55,F8.2)
      K=41
      WRITE(NWRITE,500) K,DESIGN(K)
  500 FORMAT(T2,I2,T5,'TRASH VOLUME(FT3/MAN-DAY)                      ',
     * T55,F8.2)
      K=42
      WRITE(NWRITE,510) K,DESIGN(K)
  510 FORMAT(T2,I2,T5,'TRASH COMPACTION RATIO(FRACTION OF ORIGINAL)    ',
     * T55,F8.2)
      K=43
      WRITE(NWRITE,520) K,DESIGN(K)
  520 FORMAT(T2,I2,T5,'TRASH COMPACTOR WEIGHT(LB)                      ',
     * T55,F8.2)
      K=44
      WRITE(NWRITE,530) K,DESIGN(K)
  530 FORMAT(T2,I2,T5,'TRASH COMPACTOR VOLUME(FT3)                     ',
     * T55,F8.2)
      K=45
      WRITE(NWRITE,540) K,DESIGN(K)
  540 FORMAT(T2,I2,T5,'TRASH COMPACTOR POWER(WATTS)                    ',
```

```
     * T55,F8.2)
      K=46
      WRITE(NWRITE,550) K,DESIGN(K)
  550 FORMAT(T2,I2,T5,'CLOTHES WASHER WEIGHT(LB)                    ',
     * T55,F8.2)
      K=47
      WRITE(NWRITE,560) K,DESIGN(K)
  560 FORMAT(T2,I2,T5,'CLOTHES WASHER VOLUME(FT3)                   ',
     * T55,F8.2)
      K=48
      WRITE(NWRITE,570) K,DESIGN(K)
  570 FORMAT(T2,I2,T5,'CLOTHES WASHER POWER(WATTS)                  ',
     * T55,F8.2)
      K=49
      WRITE(NWRITE,580) K,DESIGN(K)
  580 FORMAT(T2,I2,T5,'DISHWASHER WEIGHT(LB)                        ',
     * T55,F8.2)
      K=50
      WRITE(NWRITE,590) K,DESIGN(K)
  590 FORMAT(T2,I2,T5,'DISHWASHER VOLUME(FT3)                       ',
     * T55,F8.2)
      K=51
      WRITE(NWRITE,600) K,DESIGN(K)
  600 FORMAT(T2,I2,T5,'DISHWASHER POWER(WATTS)                      ',
     * T55,F8.2)
      K=52
      WRITE(NWRITE,610) K,DESIGN(K)
  610 FORMAT(T2,I2,T5,'POWER PENALTY,LIGHT SIDE(LB/WATT)            ',
     * T55,F8.2)
      K=53
      WRITE(NWRITE,615) K,DESIGN(K)
  615 FORMAT(T2,I2,T5,'POWER PENALTY,DARK SIDE(LB/WATT)             ',
     * T55,F8.2)
      K=54
      WRITE(NWRITE,620) K,DESIGN(K)
  620 FORMAT(T2,I2,T5,'AVG HEAT REJECTION PENALTY(LB/WATT)          ',
     * T55,F8.2)
      K=55
      WRITE(NWRITE,630) K,DESIGN(K)
  630 FORMAT(T2,I2,T5,'TIME IN SUNLIGHT PER ORBIT(MIN)              ',
     * T55,F8.2)
      K=56
      WRITE(NWRITE,640) K,DESIGN(K)
  640 FORMAT(T2,I2,T5,'TIME IN DARKNESS PER ORBIT(MIN)              ',
     * T55,F8.2)
C     CALL DSGNIN(*445)
      IF (NWRITE.EQ.6) CALL DSGNIN(*445)
      RETURN
      END
C
      SUBROUTINE DSGNIN(*)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
```

```
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
      LOGICAL FLAG
      FLAG=.FALSE.
    5 PRINT*,'ENTER ID#,SPACE,NEW VALUE <RET> FOR EACH CHANGE'
      PRINT*,'ENTER 0,SPACE,0 <RET> TO QUIT EDITING'
   10 READ(5,*,ERR=5) NUM,VALUE
      IF (NUM.EQ.0) GOTO 99
      IF (NUM.LT.1.OR.NUM.GT.NDSGN) GOTO 10
      DESIGN(NUM)=VALUE
      FLAG=.TRUE.
      GOTO 10
   99 IF (FLAG) RETURN 1
      RETURN
      END
C
      SUBROUTINE SSDTIN(*,ISYS)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
      LOGICAL FLAG
      FLAG=.FALSE.
    5 PRINT*,'ENTER ID#,SPACE,NEW VALUE <RET> FOR EACH CHANGE'
      PRINT*,'ENTER 0,SPACE,0 <RET> TO QUIT EDITING'
   10 READ(5,*,ERR=5) NUM,VALUE
      IF (NUM.EQ.0) GOTO 99
      IF (NUM.LT.1.OR.NUM.GT.NSSDAT) GOTO 10
      SSDATA(NUM,ISYS)=VALUE
      FLAG=.TRUE.
      GOTO 10
   99 IF (FLAG) RETURN 1
      RETURN
      END
C
      SUBROUTINE SBRSSDAT(NWRITE)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /FILE/ FNAME(NSYS)
```

```
      CHARACTER*12 FNAME,FILNAM(NNUM)
      CHARACTER*6 ADUMP
      LOGICAL FLAG(NSYS)
      DO 3 I=1,50
    3 FLAG(I)=.FALSE.
      DO 500 I=1,NUMBER
      ISYS=ITYPE(I,1)
      IF (FLAG(ISYS)) GOTO 500
      FLAG(ISYS)=.TRUE.
      FILNAM(I)=FNAME(ISYS)
      NREAD=5
    5 WRITE(NWRITE,10) FILNAM(I)
   10 FORMAT(////,T20,'SUBSYSTEM DATA - ',A12)
      K=1
      WRITE(NWRITE,20) K,SSDATA(K,ISYS)
   20 FORMAT(T2,I2,T6,'POWER REQUIRED,DAYLIGHT SIDE(WATTS):',T44,F8.2)
      K=2
      WRITE(NWRITE,30) K,SSDATA(K,ISYS)
   30 FORMAT(T2,I2,T6,'POWER REQUIRED,DARK SIDE(WATTS):    ',T44,F8.2)
      K=3
      WRITE(NWRITE,40) K,SSDATA(K,ISYS)
   40 FORMAT(T2,I2,T6,'HEAT REJECTION(WATTS):              ',T44,F8.2)
      K=4
      WRITE(NWRITE,50) K,SSDATA(K,ISYS)
   50 FORMAT(T2,I2,T6,'INITIAL FLIGHT UNIT(LB):            ',T44,F8.2)
      K=5
      WRITE(NWRITE,60) K,SSDATA(K,ISYS)
   60 FORMAT(T2,I2,T6,'INITIAL FLIGHT UNIT(FT3):           ',T44,F8.2)
      K=6
      WRITE(NWRITE,70) K,SSDATA(K,ISYS)
   70 FORMAT(T2,I2,T6,'SPARES OVER RESUPPLY INTERVAL(LB):  ',T44,F8.2)
      K=7
      WRITE(NWRITE,80) K,SSDATA(K,ISYS)
   80 FORMAT(T2,I2,T6,'SPARES OVER RESUPPLY INTERVAL(FT3): ',T44,F8.2)
      K=8
      WRITE(NWRITE,90) K,SSDATA(K,ISYS)
   90 FORMAT(T2,I2,T6,'CONSUMMABLES OVER RESUPPLY INT(LB): ',T44,F8.2)
      K=9
      WRITE(NWRITE,100) K,SSDATA(K,ISYS)
  100 FORMAT(T2,I2,T6,'CONSUMMABLES OVER RESUPPLY INT(FT3):',T44,F8.2)
      K=10
      WRITE(NWRITE,110) K,SSDATA(K,ISYS)
  110 FORMAT(T2,I2,T6,'RELIABILITY (1 TO 8):              ',T44,F8.2)
      K=11
      WRITE(NWRITE,120) K,SSDATA(K,ISYS)
  120 FORMAT(T2,I2,T6,'TECHNOLOGY READINESS (1 TO 8):     ',T44,F8.2)
      K=12
      WRITE(NWRITE,130) K,SSDATA(K,ISYS)
  130 FORMAT(T2,I2,T6,'PACING TECHNOLOGY PROBLEMS (1 TO 8):',T44,F8.2)
      K=13
      WRITE(NWRITE,140) K,SSDATA(K,ISYS)
  140 FORMAT(T2,I2,T6,'SAFETY INDEX (1 TO 8):             ',T44,F8.2)
      K=14
      WRITE(NWRITE,150) K,SSDATA(K,ISYS)
  150 FORMAT(T2,I2,T6,'MAINTAINABILITY INDEX (1 TO 8):    ',T44,F8.2)
      K=15
      WRITE(NWRITE,160) K,SSDATA(K,ISYS)
  160 FORMAT(T2,I2,T6,'COMMONALITY INDEX (1 TO 8):        ',T44,F8.2)
      K=16
      WRITE(NWRITE,170) K,SSDATA(K,ISYS)
```

```
170 FORMAT(T2,I2,T6,'MAINTENANCE OVER RESUPPLY INT(HR):   ',T44,F8.2)
    IF (ISYS.GE.21.AND.ISYS.LE.30) THEN
      K=18
      WRITE(NWRITE,190) K,SSDATA(K,ISYS)
190   FORMAT(T2,I2,T6,'LB/DAY O2 USED TO SIZE SYSTEM:       ',T44,F8.2)
    ENDIF
    IF (ISYS.GE.1.AND.ISYS.LE.20) THEN
      K=19
      WRITE(NWRITE,200) K,SSDATA(K,ISYS)
200   FORMAT(T2,I2,T6,'LB/DAY CO2 USED TO SIZE SYSTEM:      ',T44,F8.2)
    ENDIF
    IF (ISYS.GE.31) THEN
      K=20
      WRITE(NWRITE,210) K,SSDATA(K,ISYS)
210   FORMAT(T2,I2,T6,'LB/DAY H2O USED TO SIZE SYSTEM:      ',T44,F8.2)
    ENDIF
    IF(NWRITE.EQ.6) CALL SSDTIN(*5,ISYS)
215 IF (ISYS.EQ.1) THEN
      K=21
      WRITE(NWRITE,220) K,SSDATA(K,ISYS)
220   FORMAT(T2,I2,T6,'H2 USED OVER RESUPPLY INTERVAL(LBM):',T44,F8.2)
      K=22
      WRITE(NWRITE,230) K,SSDATA(K,ISYS)
230   FORMAT(T2,I2,T6,'O2 USED OVER RESUPPLY INTERVAL(LBM):',T44,F8.2)
    ENDIF
    IF (ISYS.GE.11.AND.ISYS.LE.13) THEN
      K=21
      WRITE(NWRITE,220) K,SSDATA(K,ISYS)
    ENDIF
    IF (ISYS.GE.1.AND.ISYS.LE.20) THEN
      K=23
      WRITE(NWRITE,240) K,SSDATA(K,ISYS)
240   FORMAT(T2,I2,T6,'H2O PRODUCED OVER RESUPPLY INT(LBM):',T44,F8.2)
    ENDIF
    IF (ISYS.EQ.11.OR.ISYS.EQ.13) THEN
      K=24
      WRITE(NWRITE,250) K,SSDATA(K,ISYS)
250   FORMAT(T2,I2,T6,'C PRODUCED OVER RESUPPLY INT(LBM):   ',T44,F8.2)
    ENDIF
    IF (ISYS.EQ.12) THEN
      K=25
      WRITE(NWRITE,260) K,SSDATA(K,ISYS)
260   FORMAT(T2,I2,T6,'CH4 PRODUCED OVER RESUPPLY INT(LBM):',T44,F8.2)
      K=26
      WRITE(NWRITE,270) K,SSDATA(K,ISYS)
270   FORMAT(T2,I2,T6,'CO2 NOT REACTED OVER RESUPLY INT(LB)',T44,F8.2)
    ENDIF
    K=27
    WRITE(NWRITE,280) K,SSDATA(K,ISYS)
280 FORMAT(T2,I2,T6,'SIZING FACTOR FOR WEIGHTS            ',T44,F8.2)
    K=28
    WRITE(NWRITE,290) K,SSDATA(K,ISYS)
290 FORMAT(T2,I2,T6,'SIZING FACTOR FOR VOLUMES            ',T44,F8.2)
    K=29
    WRITE(NWRITE,300) K,SSDATA(K,ISYS)
300 FORMAT(T2,I2,T6,'SIZING FACTOR FOR POWER/HEAT REJECT ',T44,F8.2)
    K=30
    WRITE(NWRITE,310) K,SSDATA(K,ISYS)
310 FORMAT(T2,I2,T6,'E,PROCESS EFFICIENCY:                ',T44,F8.2)
    IF (NWRITE.EQ.6) CALL SSDTIN(*215,ISYS)
```

```
  500 CONTINUE
      RETURN
      END
C
      SUBROUTINE SBRAUXIL(NWRITE)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WR  X(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      DIMENSION AUXATOT(NNUM,3),AUXWTOT(NNUM,3)
      COMMON /FILE/ FNAME(NSYS)
      CHARACTER*12 FNAME,FILNAM(NNUM)
      CHARACTER*22 NAMEA(NAAUX),NAMEW(0:NWAUX)
      CHARACTER*61 WATTIT
      CHARACTER*6 ADUMP
      CHARACTER*3 AVAL
      DATA NAMEA/ 'WATER PUMP','HOT WATER H/X','PURIFICATION DEVICE',
     * 'O2 COMPRESSOR','O2 STORAGE TANK','H2 COMPRESSOR',
     * 'H2 STORAGE TANK','CH4/CO2 COMPRESSOR','CH4/CO2 STORAGE TANK',
     * 'CARBON STORAGE EQUIP'/
      DATA NAMEW/ 'NO.RECYCLES/RESUP INT','TANKAGE(PER LB H2O)',
     * 'PRIMARY PUMP','SECONDARY PUMP','RECIRCULATION PUMP',
     * 'AIR-H2O SEPARATOR','BIOCIDE RESERVOIR','PRETREAT RESERVOIR',
     * 'SPRING LOAD ACUMULATR','SHUT OFF VALVE','SOLENOID VALVE',
     * 'CHECK VALVE','PREFILTER PACKAGE','CARBON & ION EXCHANGE',
     * 'I2 DISPENSER/DETECTOR','SAMPLE PORT','LINE/DUCT(PER FOOT)'/
      DO 50 I=1,NUMBER
      ISYS=ITYPE(I,1)
      FILNAM(I)=FNAME(ISYS)
      IF (ISYS.LE.30) AVAL='ARS'
      IF (ISYS.GE.31) AVAL='WRS'
      WATTIT=' '
      ICNT=1
      IF (IWATER(1).EQ.I) THEN
       WATTIT(ICNT:ICNT+12)= ' CONDENSATE '
       ICNT=INCT+13
      ENDIF
      IF (IWATER(2).EQ.I) THEN
       WATTIT(ICNT:ICNT+14)=' REDUCTION H2O '
       ICNT=ICNT+15
      ENDIF
      IF (IWATER(3).EQ.I) THEN
       WATTIT(ICNT:ICNT+8)=' HYGIENE '
       ICNT=ICNT+9
      ENDIF
      IF (IWATER(4).EQ.I) THEN
       WATTIT(ICNT:ICNT+8)=' CLOTHES '
       ICNT=ICNT+9
      ENDIF
      IF (IWATER(5).EQ.I) THEN
       WATTIT(ICNT:ICNT+6)=' URINE '
       ICNT=ICNT+7
```

```
      ENDIF
      IF (IWATER(6).EQ.I) THEN
       WATTIT(ICNT:ICNT+8)=' DISHWASH'
      ENDIF
    5 WRITE(NWRITE,10) AVAL,FILNAM(I)
   10 FORMAT(8(/),T17,A3,' AUXILIARY EQUIPMENT DATA - ',A12)
      IF (ISYS.GE.31) WRITE(NWRITE,23) WATTIT
   23 FORMAT(1X,'H2O RECOVERED: ',A61)
      WRITE(NWRITE,20)
   20 FORMAT(T6,'ITEM',T26,'QUAN. WEIGHT EACH(LB) VOLUME EACH(FT3) ',
     * 'POWER EACH(WATT)')
      DO 25 K=1,3
      AUXATOT(I,K)=0.
   25 AUXWTOT(I,K)=0.
      IF (ISYS.LE.30) THEN
        DO 40 J=1,NAAUX
        WRITE(NWRITE,30) J,NAMEA(J),(AUXILA(I,J,K),K=0,3)
   30   FORMAT(1X,I2,'.',A22,T25,F6.2,T35,F8.3,T50,F9.4,T68,F7.2)
        DO 35 K=1,3
   35   AUXATOT(I,K)=AUXATOT(I,K)+AUXILA(I,J,K)*AUXILA(I,J,0)
   40   CONTINUE
      ELSE
        DO 60 J=0,NWAUX
        IF (J.EQ.0) THEN
          WRITE(NWRITE,51) J,NAMEW(J),AUXILW(I,J,0)
   51     FORMAT(1X,I2,'.',A22,T27,F5.2)
        ELSE
          WRITE(NWRITE,30) J,NAMEW(J),(AUXILW(I,J,K),K=0,3)
        ENDIF
        DO 55 K=1,3
   55   IF (J.GT.1)AUXWTOT(I,K)=AUXWTOT(I,K)+AUXILW(I,J,K)*AUXILW(I,J,0)
   60   CONTINUE
      ENDIF
      IF (ISYS.LE.30) THEN
        WRITE(NWRITE,70) (AUXATOT(I,K),K=1,3)
   70   FORMAT(T5,'TOTALS',T35,F8.3,T50,F9.4,T68,F7.2)
      ELSE
        WRITE(NWRITE,75) (AUXWTOT(I,K),K=1,3)
   75   FORMAT(T5,'TOTALS(EXCL. TANKAGE)',T35,F8.3,T50,F9.4,T68,F7.2)
      ENDIF
      IF (NWRITE.EQ.6) CALL AUXILIN(ISYS,I,*5)
   50 CONTINUE
      RETURN
      END
C
      SUBROUTINE AUXILIN(ISYS,I,*)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
      LOGICAL FLAG
```

```
      FLAG=.FALSE.
      PRINT 6
    6 FORMAT(1X,'ENTER ID#,QUANTITY,WEIGHT,VOLUME,POWER <RET> FOR EACH',
     * ' CHANGE(-1. TO KEEP VALUE)')
      PRINT*,'ENTER -1 0 0 0 0 <RET> TO QUIT EDITING'
   10 READ(5,*,ERR=10) NUM,Q,W,V,P
      IF (NUM.EQ.-1) GOTO 99
      IF (ISYS.LE.30) THEN
        IF (NUM.LT.1.OR.NUM.GT.NAAUX) GOTO 10
        IF (Q.GE.0.) AUXILA(I,NUM,0)=Q
        IF (W.GE.0.) AUXILA(I,NUM,1)=W
        IF (V.GE.0.) AUXILA(I,NUM,2)=V
        IF (P.GE.0.) AUXILA(I,NUM,3)=P
        FLAG=.TRUE.
        GOTO 10
      ELSE
        IF (NUM.LT.0.OR.NUM.GT.NWAUX) GOTO 10
        IF (Q.GE.0.) AUXILW(I,NUM,0)=Q
        IF (W.GE.0.) AUXILW(I,NUM,1)=W
        IF (V.GE.0.) AUXILW(I,NUM,2)=V
        IF (P.GE.0.) AUXILW(I,NUM,3)=P
        FLAG=.TRUE.
        GOTO 10
      ENDIF
   99 IF (FLAG) RETURN 1
      RETURN
      END
C
      SUBROUTINE AUXLSTRT
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
C
C  AUXILA(1-3,J,K) = AUXILIARY DATA FOR AIR REVITALIZATION SUBSYSTEM CHOSEN
C
C  AUXILA(I, 1,K)  = WATER PUMP
C  AUXILA(I, 2,K)  = HOT WATER HEAT EXCHANGER
C  AUXILA(I, 3,K)  = PURIFICATION DEVICE
C  AUXILA(I, 4,K)  = O2 COMPRESSOR
C  AUXILA(I, 5,K)  = O2 STORAGE TANK
C  AUXILA(I, 6,K)  = H2 COMPRESSOR
C  AUXILA(I, 7,K)  = H2 STORAGE TANK
C  AUXILA(I, 8,K)  = CH4/CO2 COMPRESSOR
C  AUXILA(I, 9,K)  = CH4/CO2 STORAGE TANK
C  AUXILA(I,10,K)  = CARBON STORAGE EQUIPMENT
C
C  AUXILA(I,J, 0)  = QUANTITY
C  AUXILA(I,J, 1)  = WEIGHT EACH (LB)
C  AUXILA(I,J, 2)  = VOLUME EACH (FT3)
C  AUXILA(I,J, 3)  = POWER EACH (WATTS)
```

```
C
C
C  AUXILW(1-9,J,K) = AUXILIARY DATA FOR EACH WATER RECLAMATION SUBSYSTEM CHOSE
C
C  AUXILW(I, 0,0)  = NUMBER OF RECYCLES H2O/RESUPPLY INTERVAL
C  AUXILW(I, 1,K)  = WATER TANKS: WEIGHTS,VOLUMES,POWER PER LB H2O STORED
C  AUXILW(I, 2,K)  = PRIMARY PUMP
C  AUXILW(I, 3,K)  = SECONDARY PUMP
C  AUXILW(I, 4,K)  = RECIRCULATION PUMP
C  AUXILW(I, 5,K)  = AIR-H2O SEPARATOR
C  AUXILW(I, 6,K)  = BIOCIDE RESERVOIR
C  AUXILW(I, 7,K)  = PRETREAT RESERVOIR
C  AUXILW(I, 8,K)  = SPRING LOADED ACCUMULATOR
C  AUXILW(I, 9,K)  = SHUT OFF VALVE
C  AUXILW(I,10,K)  = SOLENOID VALVE
C  AUXILW(I,11,K)  = CHECK VALVE
C  AUXILW(I,12,K)  = PREFILTER PACKAGE
C  AUXILW(I,13,K)  = CARBON & ION EXCHANGE
C  AUXILW(I,14,K)  = I2 DISPENSER/DETECTOR
C  AUXILW(I,15,K)  = SAMPLE PORT
C  AUXILW(I,16,K)  = LINE/DUCT: WEIGHTS,VOLUMES,POWER PER FOOT OF LENGTH
C
C  AUXILW(I,J, 0)  = QUANTITY
C  AUXILW(I,J, 1)  = WEIGHT EACH (LB)
C  AUXILW(I,J, 2)  = VOLUME EACH (FT3)
C  AUXILW(I,J, 3)  = POWER EACH (WATTS)
C
C
      DO 5 I=1,NUMBER
      DO 4 K=0,3
      DO 2 J=1,NAAUX
    2 AUXILA(I,J,K)=0.
      DO 3 J=0,NWAUX
    3 AUXILW(I,J,K)=0.
    4 CONTINUE
    5 CONTINUE
      DO 100 I=1,NUMBER
      ISYS=ITYPE(I,1)
      IF (ISYS.LE.30) THEN
        DO 20 J=1,NAAUX
        DO 10 K=0,3
        IF (K.EQ.0) THEN
          AUXILA(I,J,K)=ARSAUXQ(ISYS,J)
        ELSE
          AUXILA(I,J,K)=ARSAUX(J,X)
        ENDIF
   10   CONTINUE
   20   CONTINUE
        IF (ADUMP.EQ.'DUMPED') THEN
          AUXILA(I,6,0)=0.
          AUXILA(I,7,0)=0.
          AUXILA(I,8,0)=0.
          AUXILA(I,9,0)=0.
        ENDIF
      ELSE
        ICNT=0
        LL=I-(NUMBER-IUNIT)
   25   DO 60 JJ=1,GROUP
          LOOP=0
          DO 50 II=1,NWNUM
```

```
            IF (IWATER(II).NE.JJ) GOTO 50
            IF (JFLOW(JJ,LL).NE.0) THEN
   26         ICNT=ICNT+1
               DO 40 J=0,NWAUX
                DO 30 K=0,3
                  IF (K.EQ.0) THEN
                    IF (J.EQ.5) THEN
                      AUXILW(I,J,K)=AUXILW(I,J,K)+WRSAUXQ(II,J)
                    ELSE
                      AUXILW(I,J,K)=AMAX1(AUXILW(I,J,K),WRSAUXQ(II,J))
                    ENDIF
                  ELSE
                    AUXILW(I,J,K)=WRSAUX(J,K)
                  ENDIF
   30           CONTINUE
   40         CONTINUE
            ENDIF
            LOOP=1
   50     CONTINUE
        IF(JFLOW(JJ,LL).NE.0.AND.LOOP.NE.1) THEN
          DO 61 MM=1,GROUP
            DO 62 NN=1,NWNUM
              IF(IWATER(NN).NE.MM) GOTO 62
              DO 63 MN=1,NCNIT
                IF(LFLOW(MM,MN).NE.0) THEN
                  ICNT=ICNT+1
                  DO 64 J=0,NWAUX
                    DO 65 K=0,3
                      IF(K.EQ.0) THEN
                        IF(J.EQ.5) THEN
                          AUXILW(I,J,K)=AUXILW(I,J,K)+WRSAUXQ(NN,J)
                        ELSE
                         AUXILW(I,J,K)=AMAX1(AUXILW(I,J,K),WRSAUXQ(NN,J))
                        ENDIF
                      ELSE
                        AUXILW(I,J,K)=WRSAUX(J,K)
                      ENDIF
   65                 CONTINUE
   64               CONTINUE
                  ENDIF
   63           CONTINUE
   62         CONTINUE
   61       CONTINUE
          ENDIF
   60     CONTINUE
          AUXILW(I,16,0)=AUXILW(I,16,0)*(1.+.10*(ICNT-1))
        ENDIF
  100 CONTINUE
        RETURN
        END
C
        SUBROUTINE NUMUNIT'NUNIT,NALT)
        PRINT*,'    '
        PRINT*,'ENTER NUMBER OF UNITS TO HANDLE LOAD(EX: 2 WITH 6 CREW=',
       * ' 2 3-MAN UNITS)'
   10   READ(5,*,ERR=10, INUM
        IF (INUM.LT.0) GOTO 10
        NUNIT=INUM
        PRINT*,'ENTER NUMBER OF REDUNDANT(BACKUP) UNITS (SAME SIZE AS ',
       * 'ABOVE)'
```

```
   20 READ(5,*,ERR=20) INUM
      IF (INUM.LT.0) GOTO 20
      NALT=INUM
      RETURN
      END
C
      SUBROUTINE STORE
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(NWNUM,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILB(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),
     * COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      CHARACTER*6 ADUMP
      CHARACTER*60 COMPARD
      CHARACTER*12 COMPARL
      PRINT 5
    5 FORMAT(5(/))
      PRINT*,'DO YOU WANT TO:'
      PRINT*,'  1. STORE SUMMARY DATA FOR LATER SYSTEMS COMPARISON'
      PRINT*,'  2. RETURN TO SYSTEMS SELECTION WITHOUT SAVING RESULTS'
    6 READ(5,*,ERR=6) INUM
      IF (INUM.NE.1.AND.INUM.NE.2) GOTO 6
      IF (INUM.EQ.2) GOTO 99
      IF (ITRACK.EQ.NCOMPAR) THEN
        PRINT*,'     '
        PRINT*,'***WARNING***MAX NUMBER OF SYSTEMS HAVE BEEN STORED'
        PRINT*,' 0. DO NOT SAVE CURRENT SYSTEM RESULTS'
        DO 2 I=1,ITRACK
        PRINT 3,I,COMPARL(I)
    3   FORMAT(1X,I2,'. ',A12)
    2   CONTINUE
        PRINT*,'ENTER NUMBER OF SYSTEM TO BE REPLACED'
    4   READ(5,*,ERR=4) INUM1
        IF (INUM1.EQ.0) GOTO 99
        IF (INUM1.LT.1.OR.INUM1.GT.NCOMPAR) GOTO 4
        ITRACK1=INUM1
        GOTO 9
      ENDIF
      ITRACK=ITRACK+1
      ITRACK1=ITRACK
    9 PRINT 10
   10 FORMAT(//,1X,'ENTER 12 CHARACTER IDENTIFYING LABEL',/,1X,
     * '123456789012')
   11 READ(5,15,ERR=11) COMPARL(ITRACK1)
   15 FORMAT(A12)
      PRINT 20
   20 FORMAT(/,1X,'ENTER 60 CHARACTER DESCRIPTION',/,1X,
     * 6('1234567890'))
   21 READ(5,25,ERR=21, COMPARD(ITRACK1)
```

```
   25 FORMAT(A60)
C INITIAL LAUNCH WEIGHT(LB)
      COMPAR( 1,ITRACK1)=TOTALS(28,3)
C RESUPPLY WEIGHT(LB)
      COMPAR( 2,ITRACK1)=TOTALS(30,3)
C EQUIVALENT WEIGHT(LB)
      COMPAR( 3,ITRACK1)=TOTALS(32,3)
C LIFETIME WEIGHT(LB)
      COMPAR( 4,ITRACK1)=TOTALS(33,3)
C TOTAL POWER REQUIRED(WATTS)
      COMPAR( 5,ITRACK1)=TOTALS(35,3)
C HEAT REJECTION(WATTS)
      COMPAR( 6,ITRACK1)=TOTALS( 3,3)
C ON ORBIT VOLUME(FT3)
      COMPAR( 7,ITRACK1)=TOTALS(29,3)
C RESUPPLY VOLUME(FT3)
      COMPAR( 8,ITRACK1)=TOTALS(31,3)
C LIFETIME VOLUME(FT3)
      COMPAR( 9,ITRACK1)=TOTALS(34,3)
C RELIABILITY (1-8)
      COMPAR(10,ITRACK1)=TOTALS(10,3)
C TECHNOLOGY READINESS (1-8)
      COMPAR(11,ITRACK1)=TOTALS(11,3)
C PACING TECHNOLOGY PROBLEMS (1-8)
      COMPAR(12,ITRACK1)=TOTALS(12,3)
C SAFETY (1-8)
      COMPAR(13,ITRACK1)=TOTALS(13,3)
C MAINTAINABILITY (1-8)
      COMPAR(14,ITRACK1)=TOTALS(14,3)
C COMMONALITY (1-8)
      COMPAR(15,ITRACK1)=TOTALS(15,3)
C OVERALL RATING (1-8)
      COMPAR(16,ITRACK1)=TOTALS(27,3)
C MAINTENANCE TIME (HR)
      COMPAR(17,ITRACK1)=TOTALS(16,3)
   99 RETURN
      END
      SUBROUTINE WRESUP(NWRITE)
      REAL N,M,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,MAKEUP,MAKEUPR,
     * LAUNL
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /MATRL/ PPN2,PPO2,PTOT,O2MET,O2EVA,O2LEAK,O2AIR,O2RPRS,
     * O2COM,O2EDC,O2TOT,H2EDC,H2EDCH,H2SAB,H2TOT,H2STOR,CO2MET,CO2EVA,
     * CO2TOT,N2LEAK,N2AIR,N2RPRS,N2COM,N2PURG,N2TOT,H2OGEN,H2OREMRQ,
     * H2ORED,C,CH4,CO2LST,CO2RED,H2STORR,CRES,CH4R,CO2LSTR,H2OREM,
     * HAND,SHOWER,CLOTHES,DISHW,DRINK,FOODP,URINFL,EXPH2O,EVAH2O,
     * H2OREQ,H2OMET,H2OFOOD,H2OPROD,H2OTOT,SWEAT,HYGIENL,FOODPL,LAUNL,
     * DISHWL,COND,REDH2O,HANDLD,SHOWLD,HYGIENLD,CLOTHLD,H2OURN,H2RED,
     * URINLD,DISHWLD,CONDR,CONDP,CONDS,REDH2OR,REDH2OP,REDH2OS,HYGIENR,
     * HYGIENP,HYGIENS,CLOTHR,CLOTHP,CLOTHS,URINR,URINP,URINS,DISHWR,
```

```
     * DISHWS,OTHERLD,OTHERR,OTHERP,OTHER3,H2OLOAD,H2OREC,POTREC,POTREQ,
     * MAKEUP,EXCESS,H2OSTORR,MAKEUPR,EXCESSR,EVAH2OLD,DISHWP,H2OSTOR,
     * EXPH2OL,H2OARS,H2REQ,FECAL,H2ONET,HYGREC
      COMMON /SSLOAD/ WATER(NWAUX),E(NWAUX,NWAUX),AUXEQ(18,0:7,3),STOREW,
     * CLOTHW,
     * CLOTHV,DISHWGHT,DISHV,TRASHW,TRASHV,AUXMLT(3)
      COMMON /FILE/ FNAME(NSYS)
      CHARACTER*12 FNAME
      CHARACTER*6 ADUMP
C
      CHARACTER*3 AMAT1,AMAT2
      NREAD=5
      IF (H2BSCH.GT.H2SAB) THEN
        AMAT1= ' H2'
        AMAT2= ' C '
      ELSE
        AMAT1= 'CO2'
        AMAT2= 'CH4'
      ENDIF
      NR=IFIX(R)
      WRITE(NWRITE,10) NR
   10 FORMAT(5(/),T11,'AUXILIARY EQUIPMENT WEIGHTS(LB) - ',I3,
     * ' DAY RESUPPLY INTERVAL')
      WRITE(NWRITE,20)
   20 FORMAT(T21,'LOAD  AUX.EQUIP  WATER    TANKAGE ',
     * '  MATERIAL STORAGE-  TOTAL')
      WRITE(NWRITE,30)
   30 FORMAT(T5,'ITEM',T20,'LB/DAY EXCL.TANK  WEIGHT    ',
     * 'WEIGHT    LOST   MATL LOST WEIGHT')
      DO 65 I=1,NUMBER
      ISYS=ITYPE(I,1)
      IF (ISYS.GE.21.AND.ISYS.LE.30) THEN
        WRITE(NWRITE,40) FNAME(ISYS),O2TOT,AUXEQ(1,1,1),AUXEQ(1,7,1)
   40   FORMAT(1X,A12,T20,F6.2,T27,F8.2,T73,F8.2)
      ENDIF
      IF (ISYS.GE.1.AND.ISYS.LE.10) THEN
        WRITE(NWRITE,40) FNAME(ISYS),CO2MET,AUXEQ(2,1,1),AUXEQ(2,7,1)
      ENDIF
      IF (ISYS.GE.11.AND.ISYS.LE.20) THEN
        WRITE(NWRITE,60) FNAME(ISYS),AMAT1,CO2TOT,AUXEQ(3,1,1),
     *   AUXEQ(3,3,1),AUXEQ(3,5,1)
   60   FORMAT(1X,A12,'(',A3,')',T20,F6.2,T27,F8.2,T56,F8.2,1X,F8.2)
        WRITE(NWRITE,61) AMAT2,AUXEQ(3,4,1),AUXEQ(3,6,1),AUXEQ(3,7,1)
   61   FORMAT(T14,'(',A3,')',T56,F8.2,1X,F8.2,1X,F7.2)
      ENDIF
   65 CONTINUE
      WRITE(NWRITE,70) AUXEQ(16,1,1),AUXEQ(16,4,1),AUXEQ(16,6,1),
     * AUXEQ(16,7,1)
   70 FORMAT(T4,'TOTALS FOR ARS',T27,F8.2,T56,F8.2,1X,F8.2,1X,F7.2)
      WRITE(NWRITE,80)
   80 FORMAT()
      ICNTW=3
      DO 95 I=1,NUMBER
      ISYS=ITYPE(I,1)
      IF (ISYS.LE.30) GOTO 95
      ICNTW=ICNTW+1
      IRECYC=IFIX(AUXILW(I,0,0))
      WRITE(NWRITE,90) FNAME(ISYS),IRECYC,SUBSYS(0,I),
     * AUXEQ(ICNTW,1,1),AUXEQ(ICNTW,0,1),AUXEQ(ICNTW,2,1),
     * AUXEQ(ICNTW,3,1),AUXEQ(ICNTW,5,1),AUXEQ(ICNTW,7,1)
```

```
 90 FORMAT(1X,A12,'(',I2,')',T20,F6.2,T27,F8.2,T37,F8.2,T47,F8.2,
    * T56,F8.2,1X,F8.2,1X,F7.2)
 95 CONTINUE
    IF (STOREW.NE.0.) THEN
      ICNTW=ICNTW+1
      WRITE(NWRITE,140) STOREW,AUXEQ(ICNTW,3,1),
    *   AUXEQ(ICNTW,5,1),AUXEQ(ICNTW,7,1)
140   FORMAT(1X,'NON-RECOVRED H2O',T20,F6.2,T56,F8.2,1X,F8.2,1X,F7.2)
    ENDIF
    WRITE(NWRITE,150)OTHERS,AUXEQ(10,3,1),AUXEQ(10,5,1),AUXEQ(10,7,1)
150 FORMAT(1X,'EVA/EXP H2O LOST',T20,F6.2,T56,F8.2,1X,F8.2,1X,F7.2)
    WRITE(NWRITE,160)MAKEUP,AUXEQ(11,0,1),AUXEQ(11,5,1),AUXEQ(11,7,1)
160 FORMAT(1X,'MAKEUP H2O',T20,F6.2,T37,F8.2,T65,F8.2,1X,F7.2)
    WRITE(NWRITE,170)EXCESS,AUXEQ(12,3,1),AUXEQ(12,5,1),AUXEQ(12,7,1)
170 FORMAT(1X,'EXCESS H2O',T20,F6.2,T56,F8.2,1X,F8.2,1X,F7.2)
    WRITE(NWRITE,200) CLOTHW,AUXEQ(13,1,1),AUXEQ(13,4,1),
    * AUXEQ(13,6,1),AUXEQ(13,7,1)
200 FORMAT(1X,'RESUPPLY CLOTHES',T20,F6.2,1X,F8.2,T56,F8.2,1X,F8.2,
    * 1X,F7.2)
    WRITE(NWRITE,210) DISHWGHT,AUXEQ(14,1,1),AUXEQ(14,4,1),
    * AUXEQ(14,6,1),AUXEQ(14,7,1)
210 FORMAT(1X,'RESUPPLY DISHES',T20,F6.2,1X,F8.2,T56,F8.2,1X,F8.2,
    * 1X,F7.2)
    WRITE(NWRITE,220)TRASHW,AUXEQ(15,1,1),AUXEQ(15,4,1),
    * AUXEQ(15,6,1),AUXEQ(15,7,1)
220 FORMAT(1X,'TRASH',T20,F6.2,1X,F8.2,T56,F8.2,1X,F8.2,1X,F7.2)
    WRITE(NWRITE,180) AUXEQ(17,1,1),AUXEQ(17,0,1),AUXEQ(17,2,1),
    * AUXEQ(17,3,1),AUXEQ(17,5,1),AUXEQ(17,7,1)
180 FORMAT(T3,'TOTALS FOR WRS',T27,F8.2,T37,F8.2,T47,F8.2,T56,F8.2,
    * 1X,F8.2,1X,F7.1)
    WRITE(NWRITE,230) AUXEQ(18,1,1),AUXEQ(18,0,1),AUXEQ(18,2,1),
    * AUXEQ(18,4,1),AUXEQ(18,6,1),AUXEQ(18,7,1)
230 FORMAT(T3,'GRAND TOTALS',T27,F8.2,T37,F8.2,T47,F8.2,T56,F8.2,1X,
    * F8.2,1X,F7.1)
    IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
  1 FORMAT(I2)
C
C
    WRITE(NWRITE,510) NR
510 FORMAT(5(/),T4,'AUXILIARY EQUIPMENT VOLUMES(FT3) & POWER(WATTS)',
    * ' - ',I3,' DAY RESUPPLY INTERVAL')
    WRITE(NWRITE,520)
520 FORMAT(T22,'AUXILIARY EQUIP  TANKAGE   MATERIAL STORAGE',
    * T71,'TOTALS')
    WRITE(NWRITE,530)
C
C
530 FORMAT(T3,'ITEM(FT3,WATTS)',T22,'VOLUME',T32,'POWER',T40
    * ,'VOLUME',T49,'VOLUME',T59,'POWER',T67,'VOLUME',T76,'POWER')
C
C
    DO 565 I=1,NUMBER
    ISYS=ITYPE(I,1)
    IF (ISYS.GE.21.AND.ISYS.LE.30) THEN
      WRITE(NWRITE,540) FNAME(ISYS),(AUXEQ(1,1,K),K=2,3),
    *   (AUXEQ(1,7,K),K=2,3)
540   FORMAT(1X,A12,T21,F7.2,2X,F7.1,T66,F7.2,1X,F7.1)
    ENDIF
    IF (ISYS.GE.1.AND.ISYS.LE.10) THEN
      WRITE(NWRITE,540) FNAME(ISYS),(AUXEQ(2,1,K),K=2,3),
```

```
     *    (AUXEQ(2,7,K),K=2,3)
          ENDIF
          IF (ISYS.GE.11.AND.ISYS.LE.20) THEN
            WRITE(NWRITE,560) FNAME(ISYS),AMAT1,(AUXEQ(3,1,K),K=2,3),
     *      (AUXEQ(3,5,K),K=2,3)
560         FORMAT(1X,A12,'(',A3,')',T21,F7.2,2X,F7.1,T48,F7.2,2X,F7.1)
            WRITE(NWRITE,561) AMAT2,(AUXEQ(3,6,K),K=2,3),
     *      (AUXEQ(3,7,K),K=2,3)
561         FORMAT(T14,'(',A3,')',T48,F7.2,2X,F7.1,2X,F7.2,1X,F7.1)
          ENDIF
565       CONTINUE
          WRITE(NWRITE,570) (AUXEQ(16,1,K),K=2,3),(AUXEQ(16,6,K),K=2,3),
     *    AUXEQ(16,7,2),AUXEQ(16,7,3)
570       FORMAT(T3,'TOTALS FOR ARS',T21,F7.2,2X,F7.1,T48,F7.2,2X,F7.1,
     *    2X,F7.2,1X,F7.1)
          WRITE(NWRITE,580)
580       FORMAT()
          ICNTW=3
          DO 595 I=1,NUMBER
          ISYS=ITYPE(I,1)
          IF (ISYS.LE.30) GOTO 595
          ICNTW=ICNTW+1
          IRECYC=IFIX(AUXILW(I,0,0))
          WRITE(NWRITE,590) FNAME(ISYS),IRECYC,
     *    (AUXEQ(ICNTW,1,K),K=2,3),AUXEQ(ICNTW,2,2),
     *    (AUXEQ(ICNTW,5,K),K=2,3),AUXEQ(ICNTW,7,K),K=2,3)
590       FORMAT(1X,A12,'(',I2,')',T21,F7.2,2X,F7.1,2X,F7.2,2X,F7.2,2X,
     *    F7.1,2X,F7.2,1X,F7.1)
595       CONTINUE
          IF (STOREW.NE.0.) THEN
            ICNTW=ICNTW+1
            WRITE(NWRITE,640) (AUXEQ(ICNTW,5,K),K=2,3),
     *      (AUXEQ(ICNTW,7,K),K=2,3)
640         FORMAT(1X,'NON-RECOVRED H2O',T48,F7.2,2X,F7.1,2X,F7.2,1X,
     *      F7.1)
          ENDIF
          WRITE(NWRITE,650) (AUXEQ(10,5,K),K=2,3),
     *    (AUXEQ(10,7,K),K=2,3)
650       FORMAT(1X,'EVA/EXP H2O LOST',T48,F7.2,2X,F7.1,2X,F7.2,1X,
     *    F7.1)
          WRITE(NWRITE,660) (AUXEQ(11,5,K),K=2,3),
     *    (AUXEQ(11,7,K),K=2,3)
660       FORMAT(1X,'MAKEUP H2O',T48,F7.2,2X,F7.1,2X,F7.2,1X,
     *    F7.1)
          WRITE(NWRITE,670) (AUXEQ(12,5,K),K=2,3),
     *    (AUXEQ(12,7,K),K=2,3)
670       FORMAT(1X,'EXCESS H2O',T48,F7.2,2X,F7.1,2X,F7.2,1X,
     *    F7.1)
          WRITE(NWRITE,700) (AUXEQ(13,1,K),K=2,3),(AUXEQ(13,6,K),K=2,3),
     *    (AUXEQ(13,7,K),K=2,3)
700       FORMAT(1X,'RESUPPLY CLOTHES',T21,F7.2,2X,F7.1,T48,F7.2,2X,
     *    F7.1,2X,F7.2,1X,F7.1)
          WRITE(NWRITE,710) (AUXEQ(14,1,K),K=2,3),(AUXEQ(14,6,K),K=2,3),
     *    (AUXEQ(14,7,K),K=2,3)
710       FORMAT(1X,'RESUPPLY DISHES',T21,F7.2,2X,F7.1,T48,F7.2,2X,
     *    F7.1,2X,F7.2,1X,F7.1)
          WRITE(NWRITE,720) (AUXEQ(15,1,K),K=2,3),(AUXEQ(15,6,K),K=2,3),
     *    (AUXEQ(15,7,K),K=2,3)
720       FORMAT(1X,'TRASH',T21,F7.2,2X,F7.1,T48,F7.2,2X,
     *    F7.1,2X,F7.2,1X,F7.1)
```

```
      WRITE(NWRITE,680) (AUXEQ(17,1,K),K=2,3),AUXEQ(17,2,2),
     * (AUXEQ(17,5,K),K=2,3),(AUXEQ(17,7,K),K=2,3)
  680 FORMAT(T3,'TOTALS FOR WRS',T21,F7.2,2X,F7.1,2X,F7.2,2X,
     * F7.2,2X,F7.1,2X,F7.2,1X,F7.1)
      WRITE(NWRITE,730) (AUXEQ(18,1,K),K=2,3),AUXEQ(18,2,2),
     * (AUXEQ(18,6,K),K=2,3),(AUXEQ(18,7,K),K=2,3)
  730 FORMAT(T3,'GRAND TOTALS',T21,F7.2,2X,F7.1,2X,F7.2,2X,F7.2,2X,
     * F7.1,2X,F7.2,1X,F7.1)
      IF (NWRITE.EQ.6) READ(NREAD,1) IDUM
      RETURN
      END
      SUBROUTINE COMPARE
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER(ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /IWRT/ IWRITE
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),
     * COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      CHARACTER*6 ADUMP
      CHARACTER*60 COMPARD,COMPARD1(NCOMPAR)
      CHARACTER*12 COMPARL,COMPARL1(NCOMPAR)
      DO 5 I=1,ICOMPAR
      DO 6 J=1,3
      COMPAR1(I,J)=0.
    6 CONTINUE
    5 CONTINUE
      PRINT 3
    3 FORMAT(5(/))
      IF (ITRACK.LT.2) THEN
        PRINT*,'TOO FEW SYSTEMS HAVE BEEN ANALIZED'
        GOTO 99
      ENDIF
      ICNT=0
      PRINT*,' 0. END SYSTEM SELECTION'
      DO 10 I=1,ITRACK
      PRINT 15,I,COMPARL(I)
   15 FORMAT(1X,I2,'. ',A12)
   10 CONTINUE
      PRINT*,'ENTER UP TO 3 SYSTEMS TO COMPARE, ONE PER LINE'
   20 READ(5,*,ERR=20) INUM
      IF (INUM.LT.0.OR.INUM.GT.ITRACK) GOTO 20
      IF (INUM.EQ.0.AND.ICNT.LT.2) THEN
        PRINT*,'TOO FEW SYSTEMS HAVE BEEN CHOSEN'
        GOTO 20
      ENDIF
      IF (INUM.EQ.0) GOTO 30
      ICNT=ICNT+1
      DO 25 I=1,ICOMPAR
      COMPARL1(ICNT)=COMPARL(INUM)
```

```
      COMPARD1(ICNT)=COMPARD(INUM)
   25 COMPAR1(I,ICNT)=COMPAR(I,INUM)
      IF (ICNT.NE.3) GOTO 20
   30 NWRITE=6
      CALL POINTASG(NWRITE,ICNT,COMPARL1,COMPARD1)
      IF (IWRITE.NE.2) THEN
        NWRITE=1
        CALL POINTASG(NWRITE,ICNT,COMPARL1,COMPARD1)
      ENDIF
   99 RETURN
      END
      SUBROUTINE POINTASG(NWRITE,ICNT,COMPARL1,COMPARD1)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(10,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      CHARACTER*6 ADUMP
      CHARACTER*60 COMPARD,COMPARD1(NCOMPAR)
      CHARACTER*12 COMPARL,COMPARL1(NCOMPAR),COMPARL2(NCOMPAR)
      CHARACTER*26 NAMES(17)
      DIMENSION BEST(ICOMPAR)
      DATA NAMES/'INITIAL LAUNCH WEIGHT(LB)','RESUPPLY WEIGHT(LB)',
     * 'EQUIVALENT WEIGHT(LB)','LIFETIME WEIGHT(LB)',
     * 'POWER REQUIRED(WATTS)','HEAT REJECTION(WATTS)',
     * 'ON ORBIT VOLUME(FT3)','RESUPPLY VOLUME(FT3)',
     * 'LIFETIME VOLUME(FT3)','RELIABILITY','TECHNOLOGY READINESS',
     * 'PACING TECHNOLOGY PROBLEMS','SAFETY','MAINTAINABILITY',
     * 'COMMONALITY','OVERALL RATING','MAINTENANCE TIME(HR)'/
      DO 4 I=1,ICNT
      DO 6 K=12,1,-1
    6 IF (COMPARL1(I)(K:K).NE.' ') GOTO 7
    7 K2=K
      DO 8 K1=12,13-K,-1
      COMPARL2(I)(K1:K1)=COMPARL1(I)(K2:K2)
    8 K2=K2-1
      DO 9 K1=1,12-K
    9 COMPARL2(I)(K1:K1)=' '
    4 CONTINUE
    2 WRITE(NWRITE,5)
    5 FORMAT(5(/))
      WRITE(NWRITE,10) (COMPARL2(I),I=1,ICNT)
   10 FORMAT(T8,'ITEM',T34,'POINTS',<ICNT>(2X,A12))
      DO 50 I=1,ICOMPAR
      WRITE(NWRITE,20) I,NAMES(I),POINT(I),(COMPAR1(I,J),J=1,ICNT)
   20 FORMAT(1X,I2,'. ',A26,2X,F6.2,<ICNT>(2X,F12.3))
   50 CONTINUE
      WRITE(NWRITE,30) POINT(0)
   30 FORMAT(T6,'TOTALS',T34,F6.2)
      IF (NWRITE.EQ.6) CALL POINTIN(*2)
```

```
      IF (POINT(0).EQ.100.) GOTO 100
      PRINT*,'***WARNING***TOTAL POINTS ASSIGNED DOES NOT EQUAL 100.'
      PRINT*,'DO YOU WANT TO:'
      PRINT*,'  1. RATIO POINTS ASSIGNED TO CURRENT ITEMS TO GIVE ',
     * 'TOTAL = 100.'
      PRINT*,'  2. RETURN TO INDIVIDUAL SPECIFICATION OF POINTS'
   55 READ(5,*,ERR=55) INUM
      IF (INUM.LT.1.OR.INUM.GT.2) GOTO 55
      IF (INUM.EQ.2) GOTO 2
      IF (POINT(0).EQ.0.) THEN
        DO 58 I=1,ICOMPAR
        POINT(I)=1.
   58   POINT(0)=POINT(0)+POINT(I)
      ENDIF
      DO 60 I=1,ICOMPAR-1
   60 POINT(I)=POINT(I)*100./POINT(0)
      POINT(0)=0.
      DO 65 I=1,ICOMPAR-1
   65 POINT(0)=POINT(0)+POINT(I)
      POINT(ICOMPAR)=100.-POINT(0)
      POINT(0)=POINT(0)+POINT(ICOMPAR)
      GOTO 2
  100 DO 110 I=1,ICOMPAR
      IF (I.LT.10.OR.I.GT.16) BEST(I)=1.E30
  110 IF (I.GE.10.AND.I.LE.16) BEST(I)=0.
      DO 120 I=1,ICOMPAR
      DO 115 J=1,ICNT
      IF (I.LT.10.OR.I.GT.16) BEST(I)=AMIN1(BEST(I),COMPAR1(I,J))
      IF (I.GE.10.AND.I.LE.16) BEST(I)=AMAX1(BEST(I),COMPAR1(I,J))
  115 CONTINUE
  120 CONTINUE
      DO 123 J=1,ICNT
  123 RATE(0,J)=0.
      DO 130 I=1,ICOMPAR
      DO 125 J=1,ICNT
C     IF (POINT(I).EQ.0.) THEN
C       RATE(I,J)=0.
C       GOTO 125
C     ENDIF
      IF (I.LT.10.OR.I.GT.16) RATE(I,J)=(1.0 - ALOG(COMPAR1(I,J)/
     * BEST(I))/ALOG(2.0))*POINT(I)
      IF (I.GE.10.AND.I.LE.16) RATE(I,J)=(1.-ALOG(BEST(I)/
     * COMPAR1(I,J))/ALOG(2.))*POINT(I)
      IF (RATE(I,J).LT.0.) RATE(I,J)=0.
      RATE(0,J)=RATE(0,J)+RATE(I,J)
  125 CONTINUE
  130 CONTINUE
      WRITE(NWRITE,5)
      WRITE(NWRITE,10) (COMPARL2(I),I=1,ICNT)
      DO 150 I=1,ICOMPAR
      WRITE(NWRITE,145) I,NAMES(I),POINT(I),(RATE(I,J),J=1,ICNT)
  145 FORMAT(1X,I2,'. ',A26,2X,F6.2,T39,<ICNT>(8X,F6.2))
  150 CONTINUE
      WRITE(NWRITE,160) POINT(0),(RATE(0,J),J=1,ICNT)
  160 FORMAT(T6,'TOTALS',T34,F6.2,T39,<ICNT>(8X,F6.2))
  162 IF (NWRITE.EQ.6) READ(5,1,ERR=162) IDUM
    1 FORMAT(I2)
      IF (NWRITE.NE.6) THEN
        DO 170 I=1,ICNT
        WRITE(NWRITE,175) COMPARL1(I),COMPARD1(I)
```

```
175   FORMAT(1X,A12,':  ',A60)
170   CONTINUE
      ENDIF
      RETURN
      END
      SUBROUTINE POINTIN(*)
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      COMMON /RATESYS/ ITRACK,COMPAR(ICOMPAR,NCOMPAR),
     * POINT(0:ICOMPAR),RATE(0:ICOMPAR,NCOMPAR),
     * COMPAR1(ICOMPAR,3)
      COMMON /CHAR2/ COMPARD(NCOMPAR),COMPARL(NCOMPAR)
      CHARACTER*6 ADUMP
      CHARACTER*60 COMPARD
      CHARACTER*12 COMPARL
      LOGICAL FLAG
      FLAG=.FALSE.
    5 PRINT*,'ENTER ID#,SPACE,NUMBER OF POINTS(0-100) TO BE ASSIGNED ',
     * 'TO EACH ITEM'
      PRINT*,'ENTER 0,SPACE,0 <RET> TO END',
     * '(ENTER -1,SPACE,VALUE TO SET ALL ITEMS=VALUE)'
   10 READ(5,*,ERR=5) NUM,VALUE
      IF (NUM.EQ.0) GOTO 90
      IF (NUM.GT.ICOMPAR) GOTO 10
      IF (VALUE.LT.0.) GOTO 10
      IF (NUM.LT.0) THEN
        DO 15 I=1,ICOMPAR
   15   POINT(I)=VALUE
      ELSE
        POINT(NUM)=VALUE
      ENDIF
      FLAG=.TRUE.
      GOTO 10
   90 IF (FLAG) THEN
        POINT(0)=0.
        DO 20 I=1,ICOMPAR
   20   POINT(0)=POINT(0)+POINT(I)
        RETURN 1
      ENDIF
      RETURN
      END
      SUBROUTINE INITL
      REAL N,M
      PARAMETER (NDSGN=56,NSSDAT=30,NSYS=50,NSUBSYS=35,NNUM=9,NWNUM=6)
      PARAMETER (NAAUX=10,NWAUX=16,NITYP=4)
      PARAMETER (ICOMPAR=17,NCOMPAR=20)
      COMMON /CHAR/ ADUMP
      COMMON /GENRL/ DESIGN(NDSGN),SSDATA(NSSDAT,NSYS),
     * SUBSYS(0:NSUBSYS,NNUM),N,R,RE,M,ITYPE(NNUM,NITYP),IWATER(NWNUM),
     * TOTALS(NSUBSYS,3),NUMBER,ARSAUX(NAAUX,3),WRSAUX(NWAUX,0:3),
```

```
     * ARSAUXQ(30,NAAUX),WRSAUXQ(NWNUM,0:NWAUX),
     * AUXILA(NNUM,NAAUX,0:3),AUXILW(NNUM,0:NWAUX,0:3),
     * GROUP,IUNIT,JFLOW(NAAUX,NCOMPAR),KFLOW(NCOMPAR,NCOMPAR),
     * SWATER(NAAUX,NAAUX),PSWATER(NAAUX),LFLOW(NCOMPAR,NCOMPAR)
      CHARACTER*6 ADUMP
      DO 20 J=0,NSUBSYS
      DO 10 I=1,NNUM
      SUBSYS(J,I)=0.
   10 CONTINUE
   20 CONTINUE
      DO 40 J=1,NNUM
      DO 30 I=1,NITYP
      ITYPE(J,I)=0
   30 CONTINUE
   40 CONTINUE
      DO 50 I=1,NWNUM
   50 IWATER(I)=0
      DO 60 I=1,NAAUX
      DO 60 J=1,NCOMPAR
      JFLOW(I,J)=0
   60 CONTINUE
      DO 70 I=1,NCOMPAR
      DO 70 J=1,NCOMPAR
      KFLOW(I,J)=0
      LFLOW(I,J)=0
   70 CONTINUE
      DO 80 I=1,NAAUX
      DO 80 J=1,NAAUX
      SWATER(I,J)=0.
   80 CONTINUE
      DO 90 I=1,NAAUX
      PSWATER(I)=0.
   90 CONTINUE
      RETURN
      END
$
```