Trim to here

PNL-3161-2
UC-70

Assessment of Effectiveness of
Geologic Isolation Systems

CIRMIS Data System

Volume 2. Program Listings

D. R. Friedrichs

January 1980

Prepared for the
Office of Nuclear Waste Isolation
under its Contract with the
U.S. Department of Energy

Pacific Northwest Laboratory
Operated for the U.S. Department of Energy
by Battelle Memorial Institute

Battelle

PNL-3161-2

NOTICE

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
Under Contract EY-76-C-06-1830

Printed in the United States of America
Available from
National Technical Information Service
United States Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151

Price: Printed Copy $____*; Microfiche $3.00

*Pages	NTIS Selling Price
001-025	$4.00
026-050	$4.50
051-075	$5.25
076-100	$6.00
101-125	$6.50
126-150	$7.25
151-175	$8.00
176-200	$9.00
201-225	$9.25
226-250	$9.50
251-275	$10.75
276-300	$11.00

PNL-3161-2
UC-70

Assessment of Effectiveness of Geologic Isolation Systems

CIRMIS DATA SYSTEM

VOLUME 2. PROGRAM LISTINGS

D. R. Friedrichs

January 1980

Prepared for the
Office of Nuclear Waste Isolation
under its Contract with the
U. S. Department of Energy

Pacific Northwest Laboratory
Richland, Washington 99352

DOE-Richland, WA

FOREWORD

The Assessment of Effectiveness of Geologic Isolation Systems (AEGIS) Program is developing and applying the methodology for assessing the far-field, long-term post-closure safety of deep geologic nuclear waste repositories. AEGIS is being performed by Pacific Northwest Laboratory (PNL) under contract with the Office of Nuclear Waste Isolation (OWNI) for the Department of Energy (DOE). One task within AEGIS is the development of methodology for analysis of the consequences (water pathway) from loss of repository containment as defined by various release scenarios.

Analysis of the long-term, far-field consequences of release scenarios requires the application of numerical codes which simulate the hydrologic systems, model the transport of released radionuclides through the hydrologic systems to the biosphere, and, where applicable, assess the radiological dose to humans.

Essentially three modeling technologies are involved in assessing the water pathway release consequence. These models are: 1) hydrologic models that define the groundwater flow field and provide water flow paths and travel times, 2) transport models that describe the movement and concentrations of the radionuclides in the flow field, and 3) dose models that determine the resultant dose burdens to individuals and/or populations. Figure i is a schematic flow diagram for the release consequence analysis.

The various input parameters required in the analysis are compiled in data systems. The data are organized and prepared by various input subroutines for utilization by the hydraulic and transport codes. The hydrologic models simulate the groundwater flow systems and provide water flow directions, rates, and velocities as inputs to the transport models. Outputs from the transport models are basically graphs of radionuclide concentration in the groundwater plotted against time. After dilution in the receiving surface-water body (e.g., lake, river, bay), these data are the input source terms for the dose models, if dose assessments are required. The dose models calculate radiation dose to individuals and populations.

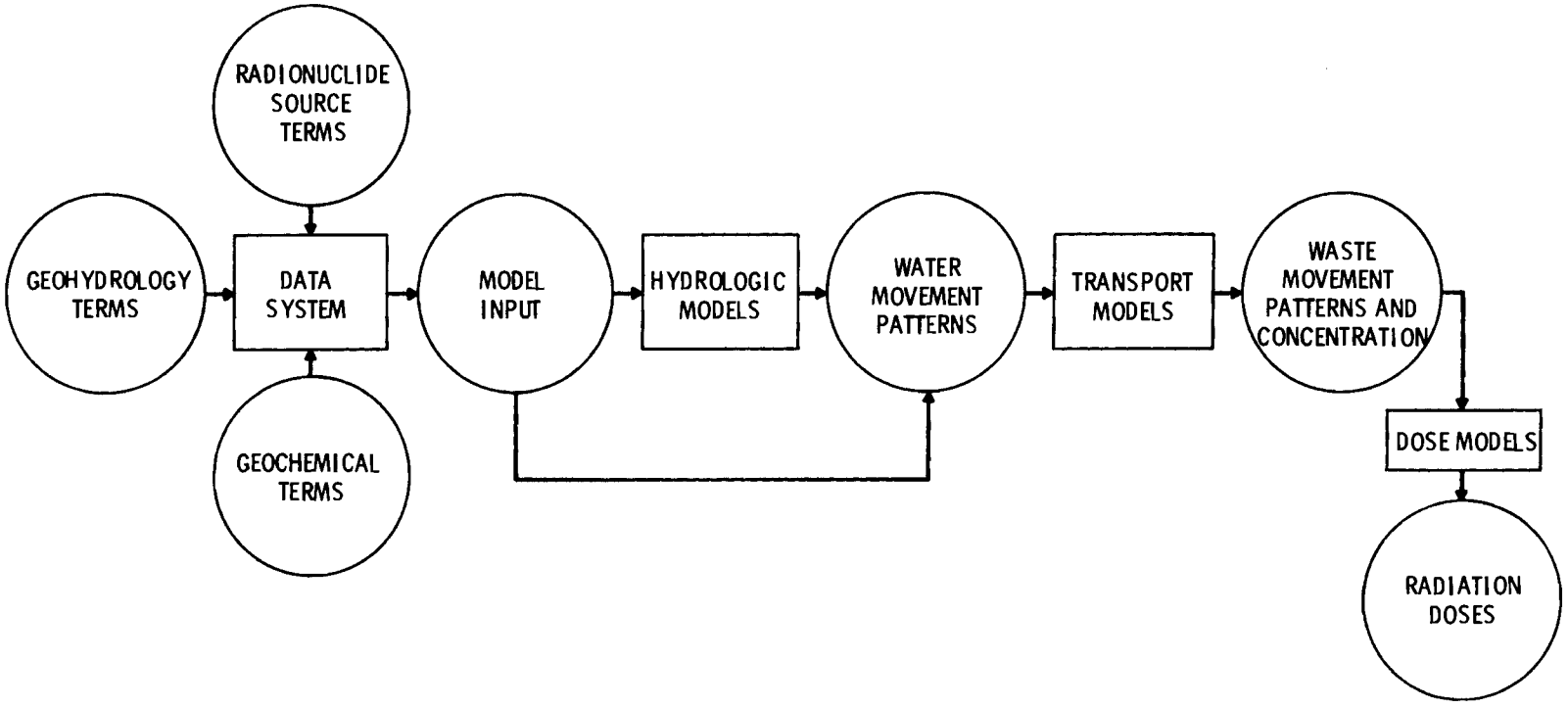

FIGURE i. Schematic Diagram of Consequence Analysis

Hydrologic and transport models are available at several levels of complexity or sophistication. Model selection and use are determined by the quantity and quality of input data. Model development under AEGIS and related programs provides three levels of hydrologic models, two levels of transport models, and one level of dose models (with several separate models). The models and data systems are documented as follows:

- HYDROLOGIC MODELS:

  PNL-3162 PATHS Groundwater Hydrologic Model - first level (simplest) idealized hybrid analytical/numerical model for two-dimensional, saturated groundwater flow and single component transport; homogeneous geology.

  PNL-3160 VTT (Variable Thickness Transient) Groundwater Hydrologic Model - second level (intermediate complexity) two-dimensional saturated groundwater flow, Boussinesq approximation, finite difference approach; two-dimensional (quasi three-dimensional) multiaquifer capability; heterogeneous geology.

  PNL-2939 FE3DGW (Finite Element, Three-Dimensional Groundwater) Hydrologic Model - third level (high complexity) three-dimensional, finite element approach (Galerkin formulation) for saturated groundwater flow; heterogeneous geology.

- TRANSPORT MODELS:

  PNL-2970 GETOUT Transport Model - first level one-dimensional analytical solution considering radioactive chain decay with capability for only simple release and hydrologic functions; single speciation, constant flow rate, dispersion and sorption, three-member straight decay chains.

  PNL-3179 MMT (Multicomponent Mass Transport) Model - second level, one-dimensional numerical, discrete parcel random walk (DPRW) algorithm; chain decay, single speciation, equilibrium sorption, time-variant leach rate and dispersion, n-membered straight or branched decay chains.

- DOSE MODELS:

  PNL-3180 ARRRG - drinking water, external exposure to aquatic food, water and shorelines, and FOOD - terrestrial food.

  PNL-3209 PABLM - Combination of ARRRG and FOOD with additional features related to chronic releases.

  BNWL-B-264 KRONIC - chronic external dose from air pathways.

  BNWL-B-351 SUBDOSA - acute external dose from air pathways.

  BNWL-B-389 DACRIN - chronic or acute inhalation dose from air pathways.

- DATA SYSTEMS:

  PNL-3139 SIRS (Sorption Information Retrieval System) - storage and retrieval system for experimental data on sorption/desorption analyses for a wide variety of radionuclides, groundwater compositions, and rocks and minerals.

  PNL-3161 CIRMIS (Comprehensive Information Retrieval and Model Input Sequence) Data System - storage and retrieval system for model input and output data, including graphical interpretation and display.

This is the second of four volumes of the description of the CIRMIS Data System.

Return of the form on the last page of this report is required in order to remain on the distribution list for future revisions of the model.

ACKNOWLEDGMENT

This research was supported by the Waste Isolation Safety Assessment Program (WISAP) conducted by Pacific Northwest Laboratory. The program was sponsored by the Office of Nuclear Waste Isolation, managed by Battelle Memorial Institute for the Department of Energy under Contract EY-76-C-06-1830. On 1 October 1979, WISAP became the Assessment of Effectiveness of Geologic Isolation Systems (AEGIS) program and the Waste/Rock Interactions Technology (WRIT) program. This report was issued by AEGIS.

CONTENTS

MNTR13.FLX

11/45 CIRMIS MONITOR (AUTOMATIC)

```
00001    C ***************  [315,100]MNTR13.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: MON-45 THIS VERSION USES NEW ABREVIATED WELL DESIGNATIONS
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C     THIS PROGRAM IS THE AUTOMATIC MONITOR ROUTINE FOR THE PDP-45
00008    C     CIRMIS NETWORK.  THIS PROGRAM INITIATES ALL REQUESTS FOR
00009    C     DATA THAT IS STORED IN THE DBØ DISK DATA BASE.
00010    C
00011    C     ALL ERROR REPORTING IS ALSO HANDLED BY THIS TASK.
00012    C
00013    C     INPUT IS FROM THE TERMINAL OR A SPECIAL COMMAND FILE MAY BE
00014    C     SET UP AS FOLLOWS:
00015    C
00016    C          LINE 1:  NUMBER OF WELLS
00017    C                   (I4)
00018    C          LINE 2:  WELL DESIGNATION
00019    C                   (12A1)
00020    C          LINE 3:  IO TYPE, IO TYPE, YEAR RANGE, NUMBER
00021    C                   OF CONTAMINATE CODES (IF IOTYP .EQ. 1)
00022    C                   (5I3)
00023    C          LINE 4:  CONTAMINATE CODES (1-N) OR "ALL", IF (IOTYP .EQ. 1)
00024    C                   (32I2) OR (A3)
00025    C
00026    C          LINES 2-4 REPEATED NWELLS TIMES, IF (IOTYP.EQ.1)
00027    C          LINE  2-3 REPEATED NWELLS TIMES, IF (IOTYP.NE.1)
00028    C
00029    C
00030    C     THE FUNCTION CODES FOR THIS PROGRAM ARE READ FROM THE DATA LINK
00031    C     AND STORED IN THE VARIBLES IDTYP AND IOTYP.
00032    C       WHERE:     IDTYP = DATA TYPE CODE (0-99999)
00033    C                  IOTYP = OUTPUT DEVICE CODE (0-99999)
00034    C
00035    C     CURRENTLY THE FOLLOWING CODES ARE DEFINED:
00036    C
00037    C     DATA TYPES:
00038    C      IDTYP = 0 --- WELL HYDROGRAPHS                 USES PEN #1
00039    C      IDTYP = 1 --- WELL CONTAMINANT HISTORIES       USES PEN #1
00040    C      IDTYP = 2 --- WELL STRUCTURES                   USES PEN #2
00041    C      IDTYP = 3 --- WELL LOG RECORDS                 USES PEN #2
00042    C      IDTYP = 4 --- SEIVE ANALYSES
00043    C      IDTYP = 5 --- PUMP TEST DATA
00044    C      IDTYP = 6 --- PHYSICAL PROPERTIES OF SOILS
00045    C      IDTYP = 7 --- CHEMICAL PROPERTIES OF SOILS
00046    C      IDTYP = 8 --- HYDROGRAPH/VTT OVERLAY
00047    C      IDTYP = 9 --- FREE
00048    C      IDTYP = 10--- WELL TEMPERATURES
00049    C      IDTYP = 11--- TRAVEL TIME PROGRAM
00050    C      IDTYP = 12--- FLOWTUBE PROGRAM
00051    C      IDTYP = 13--- PATHLINE PROGRAM
00052    C      IDTYP = 15--- SETUP HDATA FILE (TRAVEL,FLOWTUBE)
00053    C
```

```
00054   C      OUTPUT DEVICES:
00055   C       IOTYP = 0 --- UNIVAC DISPLAY
00056   C       IOTYP = 1 --- CAL-COMP PLOTTER
00057   C       IOTYP = 2 --- LINE PRINTER
00058   C       IOTYP = 3 --- GOULD PRINTER/PLOTTER
00059   C       IOTYP = 4 --- TEKTRONIX DISPLAY
00060   C       IOTYP = 5 --- ADDS PORTABLE TERMINAL
00061   C       IOTYP = 6 --- ARHCO TERMINAL
00062   C
00063   C
00064   C
00065   C  DATA FILES:
00066   C       NAME                    LUN             TYPE            ACCESS
00067   C       FILE 0--WELLHDR          1               RAN             R
00068   C       MM----TNODE              2               MM              W
00069   C       NT----DATA LINK          7               --              R/W
00070   C       COMMAND FILE             3               FOR             R
00071   C
00072   C
00073   C  LOADING SEQUENCE:
00074   C
00075   C     TKB>@[351,100]MNTR45
00076   C
00077   C
00078   C  BATTELLE MEMORIAL INSTITUTE
00079   C  PACIFIC NORTHWEST LABORATORIES
00080   C  WATER & LAND RESOURCES DEPT.
00081   C
00082   C  AUTHOR(S):     DR FRIEDRICHS
00083   C                 DW  DAMSCHEN
00084   C                 RS  ARGO
00085   C
00086   C  DATE:          INITIAL VERSION     MAY  1974
00087   C                 CURRENT VERSION    JUNE  1979
00088   C
00089   C
00090         BYTE STAT,WNAME,WLDES,REJECT(2),COMAND,ABYT(80),ANS
00091         LOGICAL ALL
00092         INTEGER WORKSP(35),PASSWD(4),STATUS(2),UIC(2),OBJTYP(2)
00093         DIMENSION IBUF(256),MSGBUF(20),SRCNOD(2),SRCTSK(2)
00094         DIMENSION FNAME(7),ICTM(32)
00095   C
00096         COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IDBIT,ICBIT(2),
00097        1            IDM(229),IDISC,IUNIT,ICNTRL,IDM2,IWNAM,IWHDR
00098         COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00099   C
00100         EQUIVALENCE (IBUF(1),WLDES(1)),(MSGBUF(1),WNAME(1))
00101         EQUIVALENCE (REJECT(1),STATUS(2))
00102   C
00103         WRITE(5,4997)
00104   4997  FORMAT(1H1,10X,'**** CIRMIS MONITOR CONTROL PROGRAM ****',//)
00105         WRITE(5,4998)
00106   4998  FORMAT(/,'SENIER FILE-0 DISC UNIT (I.E. DP1,DP2,DB0) > ')
00107         READ(5,4999) IDISC,IUNIT
00108   4999  FORMAT(A2,I1)
00109         CALL ASNLUN(1,IDISC,IUNIT)
```

```
00110          CALL DPFILE(1,'WELLHDR',6000,,IWHDR)
00111          CALL DPFILE(1,'WELLNAM',101,,IWNAM)
00112          ICNTRL = 1
00113    C
00114          WRITE(5,5000)
00115    5000  FORMAT(/,'$IS THERE A SPECIAL COMMAND FILE ? (Y OR N) > ')
00116          READ(5,5001) COMAND
00117    5001  FORMAT(A1)
00118    50    CONTINUE
00119          CONDITIONAL
00120          ,  (COMAND .EQ. 'N')
00121          .  .  READ-FROM-TI
00122          .  .  PROCESS-MONITOR-ROUTINE
00123          .  ...FIN
00124          ,  (COMAND .EQ.'Y')
00125          .  .  READ-SPECIAL-COMMAND-FILE-NAME
00126          .  .  NPLOT = 0
00127          .  .  DO (II=1,NWELLS)
00128          .  .  .  READ-DATA-FROM-COMMAND-FILE
00129          .  .  .  WHEN (IDTYP .EQ. 1)
00130          .  .  .  .  WHEN (ALL)
00131          .  .  .  .  .  DO (ICTYP=1,32)
00132    C***ALL VALUES FOR CTM 26 ARE ZERO WHICH CAUSES PLOT ROUTINES ERRORS.
00133          .  .  .  .  .  .  IF (ICTYP .NE. 26) PROCESS-MONITOR-ROUTINE
00134          .  .  .  .  .  .  ...FIN
00135          .  .  .  .  ...FIN
00136          .  .  .  .  ELSE
00137          .  .  .  .  .  DO (KK=1,NCONT)
00138          .  .  .  .  .  .  ICTYP = ICTM(KK)
00139          .  .  .  .  .  .  PROCESS-MONITOR-ROUTINE
00140          .  .  .  .  .  ...FIN
00141          .  .  .  .  ...FIN
00142          .  .  .  ...FIN
00143          .  .  .  ELSE
00144          .  .  .  .  PROCESS-MONITOR-ROUTINE
00145          .  .  .  ...FIN
00146          .  .  ...FIN
00147          .  ...FIN
00148          ...FIN
00149          WRITE(5,5002)
00150    5002  FORMAT(/,'$DO YOU WANT ANOTHER? (Y OR N) > ')
00151          READ(5,5001) ANS
00152          IF (ANS .EQ. 'Y') GO TO 50
00153          STOP

       ----------------------------------------

00154          TO PROCESS-MONITOR-ROUTINE
00155          .  NVAR  = 0
00156          .  KTEST = 0
00157          .  IF (COMAND .EQ. 'Y')
00158          .  .  NVAR    = 5
00159          .  .  KTEST   = 99
00160          .  .  IVAR(1) = IYR1
00161          .  .  IVAR(2) = 0
00162          .  .  IVAR(3) = IYR2
```

```
00163          .  .  IVAR(4) = 0
00164          .  .  IVAR(5) = 0
00165          .  .  DO (I=6,10)
00166          .  .  .  IVAR(I) = 0
00167          .  .  ...FIN
00168          .  ...FIN
00169          .  IF (IOTYP .GT. 10)
00170          .  .  IOTYP = IOTYP
00171    0     .  .  WRITE(5,511) IOTYP
00172    511   .  .  FORMAT(' IOTYP = ',I5)
00173          .  .  CALL MANSEL(IOTYP,IEGO,IOS)
00174          .  .  CHECK-FAILURES
00175          .  ...FIN
00176    C     .
00177    C     *** PROCESS A DATA REQUEST ***
00178    C     .
00179    C     ---SEARCH FOR WELL MASTER INDEX---
00180          .  CALL RTVHDR(WNAME,FQADR)
00181    C     ---WELL DATA REQUEST---
00182          .  WHEN (FQADR .GE. 0.)
00183          .  .  IGO = IOTYP
00184          .  .  IF (IOTYP .EQ. 8) IGO = 1
00185          .  .  IBUF(229) = IOTYP
00186    C     ---JUMP TO THE APPROPRIATE RETRIEVE SUBROUTINE---
00187          .  .  SELECT(IGO)
00188          .  .  .  (0)
00189    C     ***HYDROGRAPH DATA***
00190          .  .  .  .  IBIT = IFLD (0,1,IOBIT)
00191          .  .  .  .  IF (IBIT.NE.0)
00192          .  .  .  .  .  CALL HYDRTV(IEGO,IOS)
00193          .  .  .  .  .  CHECK-FAILURES
00194          .  .  .  .  .  IF (IOTYP .EQ. 1) UPDATE-PLOT-FILE
00195          .  .  .  .  ...FIN
00196          .  .  .  ...FIN
00197          .  .  .  (1)
00198    C     *** CONTAMINATE DATA ***
00199          .  .  .  .  IBIT = IFLD (1,1,IOBIT)
00200          .  .  .  .  IF (IBIT .NE. 0)
00201          .  .  .  .  .  WHEN (ICTYP .LE. 16) KBIT = IFLD(ICTYP-1,1,ICBIT(1))
00202          .  .  .  .  .  ELSE
00203          .  .  .  .  .  .  KKK  = ICTYP-17
00204          .  .  .  .  .  .  KBIT = IFLD(KKK,1,ICBIT(2))
00205          .  .  .  .  .  ...FIN
00206          .  .  .  .  .  IF (KBIT .NE. 0)
00207          .  .  .  .  .  .  IF (IOTYPE .EQ. 1) IDUM(1) = 1
00208          .  .  .  .  .  .  CALL CTMRTV(IEGO,IOS)
00209          .  .  .  .  .  .  CHECK-FAILURES
00210          .  .  .  .  .  .  IF (IOTYP .EQ. 1.AND. IDUM(1) .EQ. 1) UPDATE-PLOT-FILE
00211          .  .  .  .  .  ...FIN
00212          .  .  .  .  ...FIN
00213          .  .  .  ...FIN
00214          .  .  .  (2)
00215    C     *** WELL STRUCTURES DATA ***
00216          .  .  .  .  IBIT = IFLD (2,1,IOBIT)
00217          .  .  .  .  IF (IBIT.NE.0)
00218          .  .  .  .  .  CALL WSTRTV(IEGO,IOS)
```

```
00219          .  .  .  .  .  CHECK-FAILURES
00220          .  .  .  .  ...FIN
00221          .  .  .  ...FIN
00222          .  .  .  (3)
00223    C     *** WELL LOG RECORDS ***
00224          .  .  .  .  IBIT = IFLD (3,1,IDBIT)
00225          .  .  .  .  IF (IBIT.NE.0)
00226          .  .  .  .  .  CALL WLGRTV(IEGO,IDS)
00227          .  .  .  .  .  CHECK-FAILURES
00228          .  .  .  .  ...FIN
00229          .  .  .  ...FIN
00230          .  .  .  (4)
00231    C     *** SIEVE ANALYSIS DATA ***
00232          .  .  .  .  IBIT = IFLD (4,1,IDBIT)
00233          .  .  .  .  IF (IBIT.NE.0)
00234    C     .  .  .  .  .  CALL SVERTV(IOTYP,IEGO,IDS)
00235          .  .  .  .  .  CHECK-FAILURES
00236          .  .  .  .  ...FIN
00237          .  .  .  ...FIN
00238          .  .  .  (5)
00239    C     *** PUMP TEST DATA ***
00240          .  .  .  .  IBIT = IFLD (5,1,IDBIT)
00241          .  .  .  .  IF (IBIT.NE.0)
00242    C     .  .  .  .  .  CALL PMPRTV(IOTYP,IEGO,IDS)
00243          .  .  .  .  .  CHECK-FAILURES
00244          .  .  .  .  ...FIN
00245          .  .  .  ...FIN
00246          .  .  .  (6)
00247    C     *** PHYSICAL PROPERTIES OF SOILS DATA ***
00248          .  .  .  .  IBIT = IFLD (6,1,IDBIT)
00249          .  .  .  .  IF (IBIT.NE.0)
00250          .  .  .  .  .  CALL PHYRTV(IOTYP,IEGO,IDS)
00251          .  .  .  .  .  CHECK-FAILURES
00252          .  .  .  .  ...FIN
00253          .  .  .  ...FIN
00254          .  .  .  (7)
00255    C     *** CHEMICAL PROPERTIES OF SOILS DATA ***
00256          .  .  .  .  IBIT = IFLD (7,1,IDBIT)
00257          .  .  .  .  IF (IBIT.NE.0)
00258    C     .  .  .  .  .  CALL CHMRTV(IOTYP,IEGO,IDS)
00259          .  .  .  .  .  CHECK-FAILURES
00260          .  .  .  .  ...FIN
00261          .  .  .  ...FIN
00262          .  .  .  (10)
00263    C     *** TEMPERATURE DATA ***
00264          .  .  .  .  IBIT = IFLD (10,1,IDBIT)
00265          .  .  .  .  IF (IBIT.NE.0)
00266          .  .  .  .  .  CALL TMPRTV(IEGO,IDS)
00267          .  .  .  .  .  CHECK-FAILURES
00268          .  .  .  .  ...FIN
00269          .  .  .  ...FIN
00270          .  .  ...FIN
00271          .  ...FIN
00272    C     ---SPECIFIED WELL NOT FOUND---
00273          .  ELSE
00274          .  .  IECODE = 9999
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:08:11  PAGE  00006

 00275          .  .  WRITE(5,586) WNAME
 00276    586   .  .  FORMAT(' SPECIFIED WELL NOT FOUND > ',12A1)
 00277          .  ...FIN
 00278    C     .  WRITE(5,5998) IECODE
 00279    5998  .  FORMAT(' IECODE = ',I5)
 00280          ...FIN

        ----------------------------------------

 00281          TO READ-FROM-TI
 00282          .  WRITE(5,501)
 00283    501   .  FORMAT(/,' ENTER DESIRED DATA TYPE CODE,',/,
 00284         1.  '$     (0=HYD, 1=CTM, 2=WST, 3=WLG, 10=TMP)  > ')
 00285          .  READ (5,502) IDTYP
 00286    502   .  FORMAT(I2)
 00287          .  IF (IDTYP.EQ.1)
 00288          .  .  WRITE (5,503)
 00289    503   .  .  FORMAT(/,'$   ENTER DESIRED CONTAMINANT CODE (1-32)
 00290          .  .  READ (5,502) ICTYP
 00291          .  ...FIN
 00292          .  WRITE(5,506)
 00293    506   .  FORMAT(/,' ENTER DESIRED OUTPUT CODE,',/,
 00294         1.  '$     (1=CAL, 2=LP, 3=GLD, 4=TEK)           > ')
 00295          .  READ (5,502) IOTYP
 00296          .  WRITE (5,507)
 00297    507   .  FORMAT(/,'$ENTER WELL DESIGNATION (12A1)
 00298          .  READ (5,508) WNAME
 00299    508   .  FORMAT(12A1)
 00300          ...FIN

        ----------------------------------------

 00301          TO READ-SPECIAL-COMMAND-FILE-NAME
 00302          .  WRITE(5,5005)
 00303    5005  .  FORMAT(/,'$ENTER COMMAND FILE NAME (7A4) $')
 00304          .  READ(5,5006) FNAME
 00305    5006  .  FORMAT(7A4)
 00306          .  N = ICHR(FNAME)
 00307          .  OPEN(UNIT=3,NAME=FNAME,TYPE='OLD',READONLY)
 00308          .  READ(3,300) NWELLS
 00309    300   .  FORMAT(I4)
 00310          .  LINES = 1
 00311          .  CLOSE(UNIT=3)
 00312          ...FIN

        ----------------------------------------

 00313          TO READ-DATA-FROM-COMMAND-FILE
 00314          .  OPEN(UNIT=3,NAME=FNAME,TYPE='OLD',READONLY)
 00315          .  DO (JJ=1,LINES)
 00316          .  .  READ(3,508)
 00317          .  ...FIN
 00318          .  READ(3,508) WNAME
 00319          .  READ(3,301) IDTYP,IOTYP,IYR1,IYR2,NCONT
 00320    301   .  FORMAT(5I3)
 00321          .  IF (IDTYP .EQ. 1)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14108111  PAGE  00007

00322          .  .  ALL = .FALSE.
00323          .  .  READ (3,302) ABYT
00324   302    .  .  FORMAT(80A1)
00325          .  .  WHEN (ABYT(1).EQ.'A') ALL=.TRUE.
00326          .  .  ELSE
00327          .  .  .  III = NCONT*2
00328          .  .  .  DECODE (III,303,ABYT) (ICTM(I),I=1,NCONT)
00329   303    .  .  .  FORMAT(<NCONT>I2)
00330          .  .  ...FIN
00331          .  ...FIN
00332          .  WHEN (IDTYP .EQ. 1)
00333          .  .  LINES = LINES+3
00334          .  ...FIN
00335          .  ELSE
00336          .  .  LINES = LINES+2
00337          .  ...FIN
00338          .  CLOSE(UNIT=3)
00339          ...FIN

       ----------------------------------------

00340          TO UPDATE-PLOT-FILE
00341          .  NPLOT = NPLOT+1
00342          .  IF (NPLOT.EQ.3)
00343          .  .  CALL PLOTS (0.,0.,7)
00344          .  .  CALL PLOT (11.,-33.,3)
00345          .  .  CALL PLOTND
00346          .  .  NPLOT = 0
00347          .  ...FIN
00348          ...FIN

       ----------------------------------------

00349          TO CHECK-FAILURES
00350          .  SELECT(IEGO)
00351          .  .  (1)SUCCESSFUL-ACCESS
00352          .  .  (2)MISSING-ROUTINE-FAILURE
00353          .  .  (3)SYSTEM-DIRECTIVE-FAILURES
00354          .  ...FIN
00355          .  IF (IEGO.EQ.2. OR .IEGO.EQ.3) STOP
00356          ...FIN

       ----------------------------------------

00357          TO SUCCESSFUL-ACCESS
00358          .  IECODE = 1
00359          .  WRITE(5,580)
00360   580    .  FORMAT(' SUCCESSFUL ACCESS')
00361          ...FIN

       ----------------------------------------

00362          TO MISSING-ROUTINE-FAILURE
00363          .  IECODE = 9998
00364          .  WRITE(5,581)
00365   581    .  FORMAT(' MISSING ROUTINE FAILURE (NOT CURRENTLY IMPLEMENTED)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:08:11   PAGE  00008

  00366          ...FIN

         ----------------------------------------

  00367          TO SYSTEM-DIRECTIVE-FAILURES
  00368          ,  CONDITIONAL
  00369          .  .  (IDS .EQ. -3)
  00370          .  .  .  IECODE = 9994
  00371          .  .  .  WRITE(5,582)
  00372    582   .  .  .  FORMAT(' MEMORY NOT AVAILABLE')
  00373          .  .  ...FIN
  00374          .  .  (IDS .EQ. -2)
  00375          .  .  .  IECODE = 9995
  00376          .  .  .  WRITE(5,583)
  00377    583   .  .  .  FORMAT(' SERVICE TASK NOT INSTALLED')
  00378          .  .  ...FIN
  00379          .  .  (IDS .EQ. -7)
  00380          .  .  .  IECODE = 9996
  00381          .  .  .  WRITE(5,584)
  00382    584   .  .  .  FORMAT(' SERVICE TASK CURRENTLY ACTIVE')
  00383          .  .  ...FIN
  00384          .  .  (OTHERWISE)
  00385          .  .  .  IECODE = 9997
  00386          .  .  .  WRITE(5,585) IDS
  00387    585   .  .  .  FORMAT(' UNDIAGNOSABLE DIRECTIVE FAILURE, IDS= ',I5)
  00388          .  .  ...FIN
  00389          .  ...FIN
  00390          ...FIN
  00391          END

         ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

         00349 CHECK-FAILURES
                00174  00193  00209  00219  00227  00235  00243  00251  00259  00267

         00362 MISSING-ROUTINE-FAILURE
                00352

         00154 PROCESS-MONITOR-ROUTINE
                00122  00133  00139  00144

         00313 READ-DATA-FROM-COMMAND-FILE
                00128

         00281 READ-FROM-TI
                00121

         00301 READ-SPECIAL-COMMAND-FILE-NAME
                00125

         00357 SUCCESSFUL-ACCESS
                00351

         00367 SYSTEM-DIRECTIVE-FAILURES
```

```
        00353

      00340 UPDATE-PLOT-FILE
            00194  00210


      (FLECS VERSION 22,46)

      ----------------------------------------

00392         SUBROUTINE MANSEL(IOTYP,IEGO,IDS)
00393   C ***************  [351,100]MANSEL  ***************
00394   C
00395   C
00396   C  PROGRAM ID:
00397   C
00398   C  PROGRAM DESCRIPTION:
00399   C     THIS SUBROUTINE STARTS THE DESIRED ROUTINES
00400   C     FOR THE SPECIAL APPLICATIONS PROGRAMS.
00401   C
00402   C
00403   C
00404   C  BATTELLE MEMORIAL INSTITUTE
00405   C  PACIFIC NORTHWEST LABORATORIES
00406   C  WATER & LAND RESOURCES DEPT.
00407   C
00408   C  AUTHOR(S):      DR FRIEDRICHS
00409   C
00410   C  DATE:        INITIAL VERSION   DEC. 1974
00411   C     CURRENT VERSION   MARCH 1976
00412   C
00413   C
00414         DIMENSION IHDT(256)
00415   C
00416         COMMON/HDR/ INODE(256),IWHDR,IUNIT
00417         COMMON/HDT/ IDUM(255),KTYP
00418   C
00419         EQUIVALENCE (IDUM(1),IHDT(1))
00420   C
00421         IEGO = 1
00422   9     READ(7,710,END=10) IDUM
00423   710   FORMAT(11I5)
00424         GO TO 9
00425   10    ID = IOTYP-10
00426         IF (ID .LE. 5)
00427        .  IF (ID .EQ. 0) TASK = RAD50('DATSEL')
00428        .  IF (ID .EQ. 5) GO TO 1050
00429        .  CALL ASNLUN(1,'DB',IUNIT,STAT)
00430        .  CALL OPFILE(1,'HDATA',10000,,IPT1)
00431        .  CALL OPR(IPT1,0,,IHDT,256,)
00432        .  KTYP = IOTYP
00433        .  CALL OPW(IPT1,0,,IHDT,256,)
00434        .  CONDITIONAL
00435        .  .  (ID .EQ. 1) TASK = RAD50('TRAVEL')
00436        .  .  (ID .EQ. 2) TASK = RAD50('FLOWTR')
00437        .  .  (ID .EQ. 3) TASK = RAD50('TRAVEL')
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:08:11  PAGE  0010

 00438   1050  .  .  (ID .EQ. 5) TASK = RAD50('HDATA')
 00439         .  ...FIN
 00440         .  CALL REQUES(TASK,,IDS)
 00441         .  IF (IDS .LE. 0) GO TO 12
 00442         .  WRITE(2) TASK,IUNIT
 00443         .  RETURN
 00444         ...FIN
 00445         IEGO = 2
 00446         RETURN
 00447   12    IEGO = 3
 00448         RETURN
 00449         END

         (FLECS VERSION 22,46)

         ----------------------------------------

 00450               SUBROUTINE HYDRTV(IEGO,IDS)
 00451   C ***************  [351,100]HYDRTV.FLX  ***************
 00452   C
 00453   C
 00454   C  PROGRAM ID: CIRMIS-11-MON-SUB
 00455   C
 00456   C  PROGRAM DESCRIPTION:
 00457   C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
 00458   C     ROUTINE FOR ANY HYDROGRAPH DATA REQUEST.
 00459   C
 00460   C
 00461   C  BATTELLE MEMORIAL INSTITUTE
 00462   C  PACIFIC NORTHWEST LABORATORIES
 00463   C  WATER & LAND RESOURCES DEPT.
 00464   C
 00465   C  AUTHOR(S):   DR FRIEDRICHS
 00466   C     SW AHLSTROM
 00467   C
 00468   C  DATE:         INITIAL VERSION    MAY   1974
 00469   C     CURRENT VERSION   JUNE   1979
 00470   C
 00471   C
 00472               BYTE WNAME
 00473   C
 00474               COMMON/HDR/ INODE(258)
 00475               COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
 00476   C
 00477               IEGO = 1
 00478               IF (NVAR .GT. 0)
 00479               .  DO (K=1,NVAR)
 00480               .  .  INODE(K+203) = IVAR(K)
 00481               .  ...FIN
 00482               ...FIN
 00483               IF (IOTYP .NE. 5)
 00484               .  INODE(200) = 6
 00485               .  IF (KTEST .EQ. 99) NVAR = 99
 00486               .  INODE(203) = NVAR
 00487               .  INODE(201) = IOTYP
 00488               .  CONDITIONAL
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:08:11  PAGE  00011

00489          .  .  (IOTYP .EQ. 0) TASK = RAD50('HYDSCP')
00490          .  .  (IOTYP .EQ. 1) TASK = RAD50('HYDCAL')
00491          .  .  (IOTYP .EQ. 2) TASK = RAD50('HYDLPR')
00492          .  .  (IOTYP .EQ. 3) TASK = RAD50('HYDGLD')
00493          .  .  (IOTYP .EQ. 4) TASK = RAD50('HYDTGP')
00494          .  .  (IOTYP .EQ. 6) TASK = RAD50('HYDARC')
00495          .  ...FIN
00496          .  IEFN = 33
00497          .  CALL CLREF (IEFN,IDDS)
00498          .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IDS)
00499          .  WHEN (IDS .LT. 0) IEGO = 3
00500          .  ELSE
00501          .  .  CALL WAITFR (IEFN,IDDS)
00502          .  .  CALL WAIT(2,2,MMM)
00503          .  .  CALL CLREF (IEFN,IDDS)
00504          .  .  CALL CLREF (IEFN2,IDDS)
00505          .  ...FIN
00506          .  RETURN
00507          ...FIN
00508          IEGO = 2
00509          RETURN
00510          END

         (FLECS VERSION 22,46)

         ------------------------------------

00511               SUBROUTINE TMPRTV(IEGO,IDS)
00512    C ***************  [351,100]TMPRTV.FLX  ***************
00513    C
00514    C
00515    C  PROGRAM ID: CIRMIS-11-MON-SUB
00516    C
00517    C  PROGRAM DESCRIPTION:
00518    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00519    C     ROUTINE FOR ANY TEMPERATURE DATA REQUEST.
00520    C
00521    C
00522    C  BATTELLE MEMORIAL INSTITUTE
00523    C  PACIFIC NORTHWEST LABORATORIES
00524    C  WATER & LAND RESOURCES DEPT.
00525    C
00526    C  AUTHOR(S):    DR FRIEDRICHS
00527    C     RS ARGO
00528    C
00529    C  DATE:         INITIAL VERSION  DECEMBER 5, 1977
00530    C     CURRENT VERSION    JUNE       1979
00531    C
00532    C
00533               BYTE WNAME
00534    C
00535               COMMON/HDR/ INODE(256)
00536               COMMON/TND/ WNAME(12),IOTYP,IOTYP,ICTYP,NVAR,IVAR(10),KTEST
00537    C
00538               IEGO = 1
00539               IF (NVAR .GT. 0)
```

```
00540          .  DO (K=1,NVAR)
00541          .  .  INODE(K+203) = IVAR(K)
00542          .  ...FIN
00543          ...FIN
00544          IF (IOTYP .NE. 5)
00545          .  INODE(200) = 6
00546          .  IF (KTEST .EQ. 99) NVAR = 99
00547          .  INODE(203) = NVAR
00548          .  INODE(201) = IOTYP
00549          .  CONDITIONAL
00550          .  .  (IOTYP .EQ. 0) TASK = RAD50('TMPSCP')
00551          .  .  (IOTYP .EQ. 1) TASK = RAD50('TMPCAL')
00552          .  .  (IOTYP .EQ. 2) TASK = RAD50('TMPLPR')
00553          .  .  (IOTYP .EQ. 3) TASK = RAD50('TMPGLD')
00554          .  .  (IOTYP .EQ. 4) TASK = RAD50('TMPTOP')
00555          .  .  (IOTYP .EQ. 6) TASK = RAD50('TMPARC')
00556          .  ...FIN
00557          .  IEFN = 33
00558          .  CALL CLREF(IEFN,IDDS)
00559          .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IOS)
00560          .  WHEN (IOS .LT. 0) IEGO = 3
00561          .  ELSE
00562          .  .  CALL WAITFR (IEFN,IDDS)
00563          .  .  CALL WAIT(2,2,MMM)
00564          .  .  CALL CLREF (IEFN,IDDS)
00565          .  .  CALL CLREF (IEFN2,IDDS)
00566          .  ...FIN
00567          .  RETURN
00568          ...FIN
00569          IEGO = 2
00570          RETURN
00571          END
```

(FLECS VERSION 22.46)

```
00572                SUBROUTINE CTMRTV(IEGO,IDS)
00573     C ***************  [351,100]CTMRTV.FLX  ***************
00574     C
00575     C
00576     C  PROGRAM ID: CIRMIS-11-MON-SUB
00577     C
00578     C  PROGRAM DESCRIPTION:
00579     C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00580     C     ROUTINE FOR ANY CONTAMINANT DATA REQUEST.
00581     C
00582     C
00583     C  BATTELLE MEMORIAL INSTITUTE
00584     C  PACIFIC NORTHWEST LABORATORIES
00585     C  WATER & LAND RESOURCES DEPT.
00586     C
00587     C  AUTHOR(S):   DR FRIEDRICHS
00588     C     SW AHLSTROM
00589     C
00590     C  DATE:         INITIAL VERSION  JUNE   1974
```

```
00591    C      CURRENT VERSION  JUNE   1979
00592    C
00593    C
00594    C
00595           BYTE WNAME
00596    C
00597           COMMON/HDR/ INODE(258)
00598           COMMON/IND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00599    C
00600           IEGO = 1
00601           IF (IOTYP .EQ. 6) GO TO 6
00602           INODE(202) = ICTYP
00603           IF (NVAR .GT. 0)
00604           .  DO (K=1,NVAR)
00605           .  .  INODE(K+203) = IVAR(K)
00606           .  ...FIN
00607           ...FIN
00608           IF (IOTYP .LE. 4)
00609           .  INODE(200) = 6
00610           .  IF (KTEST .EQ. 99) NVAR = 99
00611           .  INODE(203) = NVAR
00612           .  INODE(208) = 1
00613           .  CONDITIONAL
00614           .  .  (IOTYP .EQ. 0) TASK = RAD50('CTMSCP')
00615           .  .  (IOTYP .EQ. 1) TASK = RAD50('CTMCAL')
00616           .  .  (IOTYP .EQ. 2) TASK = RAD50('CTMLPR')
00617           .  .  (IOTYP .EQ. 3) TASK = RAD50('CTMGLD')
00618           .  .  (IOTYP .EQ. 4) TASK = RAD50('CTMTGP')
00619           .  ...FIN
00620           .  GO TO 7
00621    6      .  TASK = RAD50('CTMARC')
00622    7      .  IEFN = 33
00623           .  CALL CLREF (IEFN,IDDS)
00624           .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IDS)
00625           .  WHEN (IDS .LT. 0) IEGO = 3
00626           .  ELSE
00627           .  .  CALL WAITFR (IEFN,IDDS)
00628           .  .  CALL WAIT(2,2,MMM)
00629           .  .  CALL CLREF (IEFN,IDDS)
00630           .  .  CALL CLREF (IEFN2,IDDS)
00631           .  ...FIN
00632           .  RETURN
00633           ...FIN
00634           IEGO = 2
00635           RETURN
00636           END
```

(FLECS VERSION 22,46)

```
00637           SUBROUTINE WSTRTV(IEGO,IDS)
00638    C ***************  [351,140]WSTRTV.FLX  ****************
00639    C
00640    C
00641    C  PROGRAM ID: CIRMIS-11-MON-SUB
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:08:11  PAGE  00014

00642   C
00643   C  PROGRAM DESCRIPTION:
00644   C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00645   C     ROUTINE FOR ANY WELL DOCUMENTATION DATA REQUEST.
00646   C
00647   C
00648   C  BATTELLE MEMORIAL INSTITUTE
00649   C  PACIFIC NORTHWEST LABORATORIES
00650   C  WATER & LAND RESOURCES DEPT.
00651   C
00652   C  AUTHOR(S):   DR FRIEDRICHS
00653   C     DD HOSTETLER
00654   C
00655   C  DATE:         INITIAL VERSION   AUGUST 1974
00656   C     CURRENT VERSION   JUNE   1979
00657   C
00658   C
00659         BYTE WNAME
00660   C
00661         COMMON/HDR/ INODE(250)
00662         COMMON/IND/ WNAME(12),IDTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00663   C
00664         IEGO = 1
00665         IF (IDTYP .LE. 4)
00666         .  INODE(200) = 6
00667         .  CONDITIONAL
00668         .  .  (IDTYP .EQ. 1) TASK = RAD50('WSTCAL')
00669         .  .  (IDTYP .EQ. 2) TASK = RAD50('WSTLPR')
00670         .  .  (IDTYP .EQ. 3) TASK = RAD50('WSTGLD')
00671         .  .  (IDTYP .EQ. 4) TASK = RAD50('WSTTGP')
00672         .  ...FIN
00673         .  IEFN = 33
00674         .  CALL CLREF (IEFN,IDOS)
00675         .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IOS)
00676         .  WHEN (IOS .LT. 0) IEGO = 3
00677         .  ELSE
00678         .  .  CALL WAITFR (IEFN,IDOS)
00679         .  .  CALL WAIT(2,2,MMM)
00680         .  .  CALL CLREF (IEFN,IDOS)
00681         .  .  CALL CLREF (IEFN2,IDOS)
00682         .  ...FIN
00683         .  RETURN
00684         ...FIN
00685         IEGO = 2
00686         RETURN
00687         END

        (FLECS VERSION 22,46)

        ----------------------------------------

00688         SUBROUTINE WLGRTV(IEGO,IOS)
00689   C ***************  [351,100]WLGRTV.FLX  ***************
00690   C
00691   C
00692   C  PROGRAM ID: CIRMIS-11-MON-SUB
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:08:11  PAGE  00015

00693   C
00694   C  PROGRAM DESCRIPTION:
00695   C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00696   C     ROUTINE FOR ANY WELL LOG DATA REQUEST.
00697   C
00698   C
00699   C  BATTELLE MEMORIAL INSTITUTE
00700   C  PACIFIC NORTHWEST LABORATORIES
00701   C  WATER & LAND RESOURCES DEPT.
00702   C
00703   C  AUTHOR(S):   OR FRIEDRICHS
00704   C
00705   C  DATE:        INITIAL VERSION   JULY 1974
00706   C     CURRENT VERSION   JUNE 1979
00707   C
00708   C
00709         BYTE WNAME
00710   C
00711         COMMON/HDR/ INODE(258)
00712         COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00713   C
00714         IEGO = 1
00715         IF (IOTYP .NE. 5)
00716         .  INODE(201) = IOTYP
00717         .  INODE(200) = 6
00718         .  CONDITIONAL
00719         .  .  (IOTYP .EQ. 0) TASK = RAD50('WLGSCP')
00720         .  .  (IUTYP .EQ. 1) TASK = RAD50('WLGCAL')
00721         .  .  (IOTYP .EQ. 2) TASK = RAD50('WLGLPR')
00722         .  .  (IOTYP .EQ. 3) TASK = RAD50('WLGGLD')
00723         .  .  (IOTYP .EQ. 4) TASK = RAD50('WLGTGP')
00724         .  .  (IOTYP .EQ. 6) TASK = RAD50('WLGARC')
00725         .  ...FIN
00726         .  IEFN = 33
00727         .  CALL CLREF (IEFN,IDDS)
00728         .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IDS)
00729         .  WHEN (IDS .LT. 0) IEGO = 3
00730         .  ELSE
00731         .  .  CALL WAITFR (IEFN,IDDS)
00732         .  .  CALL WAIT(2,2,MMM)
00733         .  .  CALL CLREF (IEFN,IDDS)
00734         .  .  CALL CLREF (IEFN2,IDDS)
00735         .  ...FIN
00736         .  RETURN
00737         ...FIN
00738         IEGO = 2
00739         RETURN
00740         END

        (FLECS VERSION 22.46)

        ----------------------------------------

00741         SUBROUTINE PHYRTV(IEGO,IDS)
00742   C ***************  [351,100]PHYRTV.FLX  ***************
00743   C
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:08:11  PAGE  00016

00744    C
00745    C  PROGRAM ID: CIRMIS-11-MON-SUB
00746    C
00747    C  PROGRAM DESCRIPTION:
00748    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00749    C     ROUTINE FOR ANY PHYSICAL PROPERTIES DATA REQUEST.
00750    C
00751    C
00752    C  BATTELLE MEMORIAL INSTITUTE
00753    C  PACIFIC NORTHWEST LABORATORIES
00754    C  WATER & LAND RESOURCES DEPT.
00755    C
00756    C  AUTHOR(S):    DR FRIEDRICHS
00757    C                DD HOSTETLER
00758    C
00759    C  DATE:         INITIAL VERSION   SEPT 1974
00760    C     CURRENT VERSION   JUNE 1979
00761    C
00762    C
00763          COMMON/HDR/ INODE(256)
00764    C
00765          IEGO = 1
00766    5     READ(7,701,END=10) IDUM
00767    701   FORMAT(1115)
00768          GO TO 5
00769    10    IF (IOTYP .LE. 2)
00770          .  INODE(200) = 6
00771          .  CONDITIONAL
00772          .  .  (IOTYP .EQ. 0) TASK = RAD50('PHYSCP')
00773          .  .  (IOTYP .EQ. 1) TASK = RAD50('PHYCAL')
00774          .  .  (IOTYP .EQ. 2) TASK = RAD50('PHYLPR')
00775          .  ...FIN
00776          .  IEFN = 33
00777          .  CALL CLREF (IEFN,IDDS)
00778          .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IOS)
00779          .  WHEN (IOS .LT. 0) IEGO = 3
00780          .  ELSE
00781          .  .  CALL WAITFR (IEFN,IDDS)
00782          .  .  CALL WAIT(2,2,MMM)
00783          .  .  CALL CLREF (IEFN,IDDS)
00784          .  .  CALL CLREF (IEFN2,IDDS)
00785          .  ...FIN
00786          .  RETURN
00787          ...FIN
00788          IEGO = 2
00789          RETURN
00790          END

         (FLECS VERSION 22,46)

         ----------------------------------------
```

MNTR45.FLX

11/45 CIRMIS MONITOR (INTERACTING)

```
(FLECS VERSION 22.46)     11-SEP-79   14:36:05  PAGE  00001

      ----------------------------------------

00002   C ***************  [351,100]MNTR45.FLX  ***************
00003   C
00004   C
00005   C  PROGRAM ID: MON-45 THIS VERSION USES NEW ABREVIATED WELL DESIGNATIONS
00006   C
00007   C  PROGRAM DESCRIPTION:
00008   C     THIS PROGRAM IS THE MAIN MONITOR ROUTINE FOR THE PDP-45 SIDE OF
00009   C     THE CIRMIS NETWORK.  THIS PROGRAM INITIATES ALL REQUESTS FOR
00010   C     DATA THAT IS STORED IN THE DBB DISK DATA BASE.  IT IS STARTED AND
00011   C     CONTROLLED FROM THE PDP-55 BY THE NT PSUEDO HANDLER. THIS PROGRAM
00012   C     IS THE ONLY CIRMIS ROUTINE ON THE PDP-45 SIDE OF THE NETWORK THAT
00013   C     IS ALLOWED TO READ FROM THE DATA LINK.
00014   C
00015   C     ALL ERROR REPORTING IS ALSO HANDLED BY THIS TASK.
00016   C
00017   C     THE FUNCTION CODES FOR THIS PROGRAM ARE READ FROM THE DATA LINK
00018   C     AND STORED IN THE VARIBLES IDTYP AND IOTYP.
00019   C      WHERE:            IDTYP = DATA TYPE CODE (0-99999)
00020   C                        IOTYP = OUTPUT DEVICE CODE (0-99999)
00021   C
00022   C     CURRENTLY THE FOLLOWING CODES ARE DEFINED:
00023   C
00024   C           DATA TYPES:
00025   C                 IDTYP = 0 --- WELL HYDROGRAPHS
00026   C                 IDTYP = 1 --- WELL CONTAMINANT HISTORIES
00027   C                 IDTYP = 2 --- WELL DOCUMENTATION
00028   C                 IDTYP = 3 --- WELL LOG RECORDS
00029   C                 IDTYP = 4 --- SEIVE ANALYSES
00030   C                 IDTYP = 5 --- PUMP TEST DATA
00031   C                 IDTYP = 6 --- PHYSICAL PROPERTIES OF SOILS
00032   C                 IDTYP = 7 --- CHEMICAL PROPERTIES OF SOILS
00033   C                 IDTYP = 8 --- HYDROGRAPH/VTT OVERLAY
00034   C                 IDTYP = 9 --- FREE
00035   C                 IDTYP = 10--- WELL TEMPERATURES
00036   C                 IDTYP = 11--- TRAVEL TIME PROGRAM
00037   C                 IDTYP = 12--- FLOWTUBE PROGRAM
00038   C                 IDTYP = 13--- PATHLINE PROGRAM
00039   C                 IDTYP = 15--- SETUP HDATA FILE (TRAVEL,FLOWTUBE)
00040   C
00041   C           OUTPUT DEVICES:
00042   C                 IOTYP = 0 --- UNIVAC DISPLAY
00043   C                 IOTYP = 1 --- CAL-COMP PLOTTER
00044   C                 IOTYP = 2 --- LINE PRINTER
00045   C                 IOTYP = 3 --- GOULD PRINTER/PLOTTER
00046   C                 IOTYP = 4 --- TEKTRONIX DISPLAY
00047   C                 IOTYP = 5 --- ADDS PORTABLE TERMINAL
00048   C                 IOTYP = 6 --- ARMCO TERMINAL
00049   C
00050   C
00051   C
00052   C  DATA FILES:
00053   C      NAME                  LUN            TYPE            ACCESS
00054   C      FILE 0--WELLHDR        1              RAN              R
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:36:05  PAGE  00002

00055   C       NT----DATA LINK         7                --              R/W
00056   C
00057   C
00058   C  LOADING SEQUENCE:
00059   C
00060   C     TKB>@[351,100]MNTR45
00061   C
00062   C
00063   C  BATTELLE MEMORIAL INSTITUTE
00064   C  PACIFIC NORTHWEST LABORATORIES
00065   C  WATER & LAND RESOURCES DEPT.
00066   C
00067   C  AUTHOR(S):     DR FRIEDRICHS
00068   C                 DW  DAMSCHEN
00069   C
00070   C  DATE:          INITIAL VERSION     MAY     1974
00071   C                 CURRENT VERSION FEBRUARY  1979
00072   C
00073   C
00074           BYTE STAT,WNAME,WLDES,REJECT(2)
00075           INTEGER WORKSP(35),PASSWD(4),STATUS(2),UIC(2),OBJTYP(2)
00076           DIMENSION IBUF(256),MSGBUF(20),SRCNOD(2),SRCTSK(2)
00077   C
00078           COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IDBIT,ICBIT(2),
00079          1            IDM(229),IDISC,IUNIT,ICNTRL,IDM2,IWNAM,IWHDR
00080           COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00081   C
00082           EXTERNAL ASTSUB
00083           EQUIVALENCE (IBUF(1),WLDES(1)),(MSGBUF(1),WNAME(1))
00084           EQUIVALENCE (REJECT(1),STATUS(2))
00085   C
00086   C
00087   C
00088   C
00089   C *** MONITOR THE DATA LINK ***
00090   C
00091           CALL NTINIT (STATUS,35,WORKSP)
00092           CALL NTCGTW (STATUS,ICON,SRCNOD,SRCTSK,OBJTYP,UIC,1,PASSWD)
00093           IF (STATUS(1).NE.1) WRITE(5,510) STATUS
00094   510     FORMAT(' ERROR IN NTCGTW--STATUS=',2I5)
00095           IF (PASSWD(1).NE.351)
00096          .  WRITE(5,5100) PASSWD(1)
00097   5100   .  FORMAT(' PASSWORD DOESNT MATCH.  ',I5)
00098          .  CALL NTCRJ (7,STATUS,ICON,SRCNOD,SRCTSK,OBJTYP,UIC)
00099          .  CALL EXIT
00100          ...FIN
00101           CALL NTCONW (7,STATUS,ICON,SRCNOD,SRCTSK,OBJTYP,UIC,,,ASTSUB)
00102   D       WRITE(5,5111) STATUS(1),REJECT
00103   5111    FORMAT(' STATUS,REJECT =',3I5)
00104           IF(STATUS(1).NE.1) STOP
00105           CALL NTRCVW (7,STATUS,40,MSGBUF)
00106   D       WRITE(5,5112) WNAME,IOTYP,IOTYP
00107   5112    FORMAT(' WNAME,IOTYP,IOTYP=',12A1,2I5)
00108           CALL NTDISW (7,STATUS)
00109   C
00110   C
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:36:05  PAGE  00003

 00111    C
 00112           IF (IDTYP .GT. 10)
 00113           .  IOTYP = IDTYP
 00114    0      .  WRITE(5,511) IOTYP
 00115    511    .  FORMAT(' IOTYP = ',I5)
 00116           .  CALL MANSEL(IOTYP,IEGO,IOS)
 00117           .  CHECK-FAILURES
 00118           .  CALL EXIT
 00119           ...FIN
 00120    C
 00121           IUNIT = 8
 00122           IDISC = 'DB'
 00123           CALL ASNLUN(1,IDISC,IUNIT)
 00124           CALL OPFILE(1,'WELLHDR',6000,,IWHDR)
 00125           CALL OPFILE(1,'WELLNAM',101,,IWNAM)
 00126           ICNTRL = 0
 00127           KTEST = 0
 00128    C
 00129    C      *** PROCESS A DATA REQUEST ***
 00130    C
 00131    C      ---SEARCH FOR WELL MASTER INDEX---
 00132           CALL RTVHDR(WNAME,FQADR)
 00133    C      ---WELL DATA REQUEST---
 00134           WHEN (FQADR .GE. 0.)
 00135           .  IGO = IDTYP
 00136           .  IF (IDTYP .EQ. 8) IGO = 1
 00137           .  IBUF(229) = IDTYP
 00138    C      ---JUMP TO THE APPROPRIATE RETRIEVE SUBROUTINE---
 00139           .  SELECT(IGO)
 00140           .  .  (0)
 00141           .  .  .  IBIT = IFLD (0,1,IDBIT)
 00142           .  .  .  IF (IBIT.NE.0)
 00143    C      ***HYDROGRAPH DATA***
 00144           .  .  .  .  CALL HYDRTV(IEGO,IOS)
 00145           .  .  .  .  CHECK-FAILURES
 00146           .  .  .  ...FIN
 00147           .  .  ...FIN
 00148           .  .  (1)
 00149    C      *** CONTAMINATE DATA ***
 00150           .  .  .  IBIT = IFLD (1,1,IDBIT)
 00151           .  .  .  IF (IBIT .NE. 0)
 00152           .  .  .  .  WHEN (ICTYP .LE. 16) KBIT = IFLD(ICTYP+1,1,ICBIT(1))
 00153           .  .  .  .  ELSE
 00154           .  .  .  .  .  KKK  = ICTYP-17
 00155           .  .  .  .  .  KBIT = IFLD(KKK,1,ICBIT(2))
 00156           .  .  .  .  ...FIN
 00157           .  .  .  .  IF (KBIT .NE. 0)
 00158           .  .  .  .  .  CALL CTMRTV(IEGO,IOS)
 00159           .  .  .  .  .  CHECK-FAILURES
 00160           .  .  .  .  ...FIN
 00161           .  .  .  ...FIN
 00162           .  .  ...FIN
 00163           .  .  (2)
 00164    C      *** WELL STRUCTURES DATA ***
 00165           .  .  .  CALL WSTRTV(IEGO,IOS)
 00166           .  .  .  CHECK-FAILURES
```

```
(FLECS VERSION 22.46)     11-SEP-79   14136105  PAGE  00004

00167          .  .  ...FIN
00168          .  .  (3)
00169    C     *** WELL LOG RECORDS ***
00170          .  .  .  CALL WLGRTV(IEGO,IOS)
00171          .  .  .  CHECK-FAILURES
00172          .  .  ...FIN
00173          .  .  (4)
00174    C     *** SIEVE ANALYSIS DATA ***
00175    C     .  .  .  CALL SVERTV(IOTYP,IEGO,IOS)
00176          .  .  .  CHECK-FAILURES
00177          .  .  ...FIN
00178          .  .  (5)
00179    C     *** PUMP TEST DATA ***
00180    C     .  .  .  CALL PMPRTV(IOTYP,IEGO,IOS)
00181          .  .  .  CHECK-FAILURES
00182          .  .  ...FIN
00183          .  .  (6)
00184    C     *** PHYSICAL PROPERTIES OF SOILS DATA ***
00185          .  .  .  CALL PHYRTV(IOTYP,IEGO,IOS)
00186          .  .  .  CHECK-FAILURES
00187          .  .  ...FIN
00188          .  .  (7)
00189    C     *** CHEMICAL PROPERTIES OF SOILS DATA ***
00190    C     .  .  .  CALL CHMRTV(IOTYP,IEGO,IOS)
00191          .  .  .  CHECK-FAILURES
00192          .  .  ...FIN
00193          .  .  (8)
00194    C     *** TEMPERATURE DATA ***
00195          .  .  .  CALL TMPRTV(IEGO,IOS)
00196          .  .  .  CHECK-FAILURES
00197          .  .  ...FIN
00198          .  ...FIN
00199          ...FIN
00200    C     ---SPECIFIED WELL NOT FOUND---
00201          ELSE
00202          .  IECODE = 9999
00203          ...FIN
00204          CALL EXIT

       ----------------------------------------

00205          TO CHECK-FAILURES
00206          .  SELECT(IEGO)
00207          .  .  (1)SUCCESSFUL-ACCESS
00208          .  .  (2)MISSING-ROUTINE-FAILURE
00209          .  .  (3)SYSTEM-DIRECTIVE-FAILURES
00210          .  ...FIN
00211          .  IF (IEGO.EQ.2. OR .IEGO.EQ.3) STOP
00212          ...FIN

       ----------------------------------------

00213          TO SUCCESSFUL-ACCESS
00214          .  IECODE = 1
00215    0     .  WRITE(5,580)
00216    580   .  FORMAT(' SUCCESSFUL ACCESS')
```

```
00217          ...FIN

          ----------------------------------------

00218          TO MISSING-ROUTINE-FAILURE
00219          .  IECODE = 9998
00220          .  WRITE(5,581)
00221     581  .  FORMAT(' MISSING ROUTINE FAILURE (NOT CURRENTLY IMPLEMENTED)')
00222          ...FIN

          ----------------------------------------

00223          TO SYSTEM-DIRECTIVE-FAILURES
00224          .  CONDITIONAL
00225          .  .  (IDS .EQ. -3)
00226          .  .  .  IECODE = 9994
00227          .  .  .  WRITE(5,582)
00228     582  .  .  .  FORMAT(' MEMORY NOT AVAILABLE')
00229          .  .  ...FIN
00230          .  .  (IDS .EQ. -2)
00231          .  .  .  IECODE = 9995
00232          .  .  .  WRITE(5,583)
00233     583  .  .  .  FORMAT(' SERVICE TASK NOT INSTALLED')
00234          .  .  ...FIN
00235          .  .  (IDS .EQ. -7)
00236          .  .  .  IECODE = 9996
00237          .  .  .  WRITE(5,584)
00238     584  .  .  .  FORMAT(' SERVICE TASK CURRENTLY ACTIVE')
00239          .  .  ...FIN
00240          .  .  (OTHERWISE)
00241          .  .  .  IECODE = 9997
00242          .  .  .  WRITE(5,585) IDS
00243     585  .  .  .  FORMAT(' UNDIAGNOSABLE DIRECTIVE FAILURE, IDS= ',I5)
00244          .  .  ...FIN
00245          .  ...FIN
00246          ...FIN
00247          END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00205 CHECK-FAILURES
                00117  00145  00159  00166  00171  00176  00181  00186  00191  00196

          00218 MISSING-ROUTINE-FAILURE
                00208

          00213 SUCCESSFUL-ACCESS
                00207

          00223 SYSTEM-DIRECTIVE-FAILURES
                00209


          (FLECS VERSION 22,46)
```

```
      ----------------------------------------

00248         SUBROUTINE MANSEL(IOTYP,IEGO,IOS)
00249   C ***************  [351,100]MANSEL  ***************
00250   C
00251   C
00252   C  PROGRAM IO:
00253   C
00254   C  PROGRAM DESCRIPTION:
00255   C     THIS SUBROUTINE STARTS THE DESIRED ROUTINES
00256   C     FOR THE SPECIAL APPLICATIONS PROGRAMS.
00257   C
00258   C
00259   C
00260   C  BATTELLE MEMORIAL INSTITUTE
00261   C  PACIFIC NORTHWEST LABORATORIES
00262   C  WATER & LAND RESOURCES DEPT.
00263   C
00264   C  AUTHOR(S):       DR FRIEDRICHS
00265   C
00266   C  DATE:         INITIAL VERSION   DEC. 1974
00267   C     CURRENT VERSION   MARCH 1976
00268   C
00269   C
00270         DIMENSION IHDT(256)
00271   C
00272         COMMON/HDR/ INODE(256),IWHDR,IUNIT
00273         COMMON/HDT/ IDUM(255),KTYP
00274   C
00275         EQUIVALENCE (IDUM(1),IHDT(1))
00276   C
00277         IEGO = 1
00278   9     READ(7,710,END=10) IDUM
00279   710   FORMAT(11I5)
00280         GO TO 9
00281   10    ID = IOTYP-10
00282         IF (ID .LE. 5)
00283        .  IF (ID .EQ. 0) TASK = RAD50('DATSEL')
00284        .  IF (ID .EQ. 5) GO TO 1050
00285        .  CALL ASNLUN(1,'DB',IUNIT,STAT)
00286        .  CALL OPFILE(1,'HDATA',10000.,IPT1)
00287        .  CALL OPR(IPT1,0.,IHDT,256.)
00288        .  KTYP = IOTYP
00289        .  CALL OPW(IPT1,0.,IHDT,256.)
00290        .  CONDITIONAL
00291        .  .  (ID .EQ. 1) TASK = RAD50('TRAVEL')
00292        .  .  (ID .EQ. 2) TASK = RAD50('FLOWTR')
00293        .  .  (ID .EQ. 3) TASK = RAD50('TRAVEL')
00294   1050 .  .  (ID .EQ. 5) TASK = RAD50('HDATA')
00295        .  ...FIN
00296        .  CALL REQUES(TASK,,IOS)
00297        .  IF (IOS .LE. 0) GO TO 12
00298        .  WRITE(2) TASK,IUNIT
00299        .  RETURN
00300        ...FIN
```

```
(FLECS VERSION 22,46)     11-SEP-79   14136105  PAGE  00007

  00301          IEGO = 2
  00302          RETURN
  00303    12    IEGO = 3
  00304          RETURN
  00305          END

         (FLECS VERSION 22,46)

         ------------------------------------

  00306          SUBROUTINE HYDRTV(IEGO,IDS)
  00307    C ***************  [351,100]HYDRTV,FLX  ***************
  00308    C
  00309    C
  00310    C  PROGRAM ID: CIRMIS-11-MON-SUB
  00311    C
  00312    C  PROGRAM DESCRIPTION:
  00313    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
  00314    C     ROUTINE FOR ANY HYDROGRAPH DATA REQUEST.
  00315    C
  00316    C
  00317    C  BATTELLE MEMORIAL INSTITUTE
  00318    C  PACIFIC NORTHWEST LABORATORIES
  00319    C  WATER & LAND RESOURCES DEPT.
  00320    C
  00321    C  AUTHOR(S):   DR FRIEDRICHS
  00322    C
  00323    C  DATE:          INITIAL VERSION    MAY   1978
  00324    C              CURRENT VERSION APRIL 1978
  00325    C
  00326    C
  00327          BYTE WNAME
  00328    C
  00329          COMMON/HDR/ INODE(258)
  00330          COMMON/TND/ WNAME(12),IDTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
  00331    C
  00332          IEGO = 1
  00333          IF (NVAR .GT. 0)
  00334          .  DO (K=1,NVAR)
  00335          .  .  INODE(K+203) = IVAR(K)
  00336          .  ...FIN
  00337          ...FIN
  00338          IF (IDTYP .NE. 5)
  00339          .  INODE(200) = 6
  00340          .  IF (KTEST .EQ. 99) NVAR = 99
  00341          .  INODE(203) = NVAR
  00342          .  INODE(201) = IDTYP
  00343          .  CONDITIONAL
  00344          .  .  (IDTYP .EQ. 0) TASK = RAD50('HYDSCP')
  00345          .  .  (IDTYP .EQ. 1) TASK = RAD50('HYDCAL')
  00346          .  .  (IDTYP .EQ. 2) TASK = RAD50('HYDLPR')
  00347          .  .  (IDTYP .EQ. 3) TASK = RAD50('HYDGLD')
  00348          .  .  (IDTYP .EQ. 4) TASK = RAD50('HYDTGP')
  00349          .  .  (IDTYP .EQ. 6) TASK = RAD50('HYDARC')
  00350          .  ...FIN
  00351          .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IDS)
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:36:05  PAGE  00008

  00352          .  CALL CLREF (IEFN2,IOS)
  00353          .  IF (IOS .GT. 0) RETURN
  00354          .  IEGO = 3
  00355          .  RETURN
  00356          ...FIN
  00357          IEGO = 2
  00358          RETURN
  00359          END

         (FLECS VERSION 22,46)

         ------------------------------------

  00360               SUBROUTINE TMPRTV(IEGO,IOS)
  00361     C ***************  [351,100]TMPRTV.FLX  ***************
  00362     C
  00363     C
  00364     C  PROGRAM ID: CIRMIS-11-MON-SUB
  00365     C
  00366     C  PROGRAM DESCRIPTION:
  00367     C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
  00368     C     ROUTINE FOR ANY TEMPERATURE DATA REQUEST.
  00369     C
  00370     C
  00371     C  BATTELLE MEMORIAL INSTITUTE
  00372     C  PACIFIC NORTHWEST LABORATORIES
  00373     C  WATER & LAND RESOURCES DEPT.
  00374     C
  00375     C  AUTHOR(S):    DR FRIEDRICHS
  00376     C                RS ARGO
  00377     C
  00378     C  DATE:         INITIAL VERSION  DECEMBER 5, 1977
  00379     C     CURRENT VERSION  JUNE 4, 1978
  00380     C
  00381     C
  00382               BYTE WNAME
  00383     C
  00384               COMMON/HDR/ INODE(258)
  00385               COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
  00386     C
  00387               IEGO = 1
  00388               IF (NVAR .GT. 0)
  00389               .  DO (K=1,NVAR)
  00390               .  .  INODE(K+203) = IVAR(K)
  00391               .  ...FIN
  00392               ...FIN
  00393               IF (IOTYP .NE. 5)
  00394               .  INODE(200) = 6
  00395               .  IF (KTEST .EQ. 99) NVAR = 99
  00396               .  INODE(203) = NVAR
  00397               .  INODE(201) = IOTYP
  00398               .  CONDITIONAL
  00399               .  .  (IOTYP .EQ. 0) TASK = RAD50('TMPSCP')
  00400               .  .  (IOTYP .EQ. 1) TASK = RAD50('TMPCAL')
  00401               .  .  (IOTYP .EQ. 2) TASK = RAD50('TMPLPR')
  00402               .  .  (IOTYP .EQ. 3) TASK = RAD50('TMPGLD')
```

```
(FLECS VERSION 22,46)     11-SEP-79   14136105  PAGE  00009

00403          ,  .  (IOTYP .EQ. 4) TASK = RAD50('TMPTGP')
00404          ,  .  (IOTYP .EQ. 6) TASK = RAD50('TMPARC')
00405          ,  ...FIN
00406          ,  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IOS)
00407          ,  CALL CLREF(IEFN2,IOOS)
00408          ,  IF (IOS .GT. 0) RETURN
00409          ,  IEGO = 3
00410          ,  RETURN
00411          ...FIN
00412          IEGO = 2
00413          RETURN
00414          END

         (FLECS VERSION 22,46)

         ----------------------------------------

00415               SUBROUTINE CTMRTV(IEGO,IOS)
00416    C ***************  [351,100]CTMRTV.FLX  ***************
00417    C
00418    C
00419    C  PROGRAM ID: CIRMIS-11-MON-SUB
00420    C
00421    C  PROGRAM DESCRIPTION:
00422    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00423    C     ROUTINE FOR ANY CONTAMINANT DATA REQUEST.
00424    C
00425    C
00426    C  BATTELLE MEMORIAL INSTITUTE
00427    C  PACIFIC NORTHWEST LABORATORIES
00428    C  WATER & LAND RESOURCES DEPT.
00429    C
00430    C  AUTHOR(S):   DR FRIEDRICHS
00431    C     SW AHLSTROM
00432    C
00433    C  DATE:        INITIAL VERSION   JUNE   1974
00434    C     CURRENT VERSION   MARCH 1978
00435    C
00436    C
00437    C
00438               BYTE WNAME
00439    C
00440               COMMON/HDR/ INODE(256)
00441               COMMON/TND/ WNAME(12),IOTYP,IOTYP,ICTYP,NVAR,IVAR(10),KTEST
00442    C
00443               IEGO = 1
00444               IF (IOTYP .EQ. 6) GO TO 6
00445               INODE(202) = ICTYP
00446               IF (NVAR .GT. 0)
00447               .  DO (K=1,NVAR)
00448               .  .  INODE(K+203) = IVAR(K)
00449               .  ...FIN
00450               ...FIN
00451               IF (IOTYP .LE. 4)
00452               .  INODE(200) = 6
00453               .  IF (KTEST .EQ. 99) NVAR = 99
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:36:05  PAGE  00010

00454          .  INODE(203) = NVAR
00455          .  INODE(200) = 1
00456          .  CONDITIONAL
00457          .  .  (IOTYP .EQ. 0) TASK = RAD50('CTMSCP')
00458          .  .  (IOTYP .EQ. 1) TASK = RAD50('CTMCAL')
00459          .  .  (IOTYP .EQ. 2) TASK = RAD50('CTMLPR')
00460          .  .  (IOTYP .EQ. 3) TASK = RAD50('CTMGLD')
00461          .  .  (IOTYP .EQ. 4) TASK = RAD50('CTMTOP')
00462          .  ...FIN
00463          .  GO TO 7
00464    6     .  TASK = RAD50('CTMARC')
00465    7     .  CONTINUE
00466          .  CALL VSNDRR (TASK,,INODE,IEFN2,255,,,IDS)
00467          .  CALL CLREF(IEFN2,IDDS)
00468          .  IF (IDS .GT. 0) RETURN
00469          .  IEGO = 3
00470          .  RETURN
00471          ...FIN
00472          IEGO = 2
00473          RETURN
00474          END

         (FLECS VERSION 22.46)

         ------------------------------------

00475               SUBROUTINE WSTRTV(IEGO,IDS)
00476    C ***************  [351,100]WSTRTV.FLX  ***************
00477    C
00478    C
00479    C  PROGRAM ID: CIRMIS-11-MON-SUB
00480    C
00481    C  PROGRAM DESCRIPTION:
00482    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00483    C     ROUTINE FOR ANY WELL DOCUMENTATION DATA REQUEST.
00484    C
00485    C
00486    C  BATTELLE MEMORIAL INSTITUTE
00487    C  PACIFIC NORTHWEST LABORATORIES
00488    C  WATER & LAND RESOURCES DEPT.
00489    C
00490    C  AUTHOR(S):   DR FRIEDRICHS
00491    C               DO HOSTETLER
00492    C
00493    C  DATE:        INITIAL VERSION   AUGUST 1974
00494    C     CURRENT VERSION   MAY 1978
00495    C
00496    C
00497               BYTE WNAME
00498    C
00499               COMMON/HDR/ INODE(256)
00500               COMMON/TND/ WNAME(12),IOTYP,IDTYP,ICTYP,NVAR,IVAR(10),KTEST
00501    C
00502               IEGO = 1
00503               IF (IOTYP .LE. 4)
00504               .  INODE(200) = 6
```

```
(FLECS VERSION 22,46)     11-SEP-79   14:36:05  PAGE  00011

00505          ,  CONDITIONAL
00506          ,  .  (IOTYP .EQ. 1) TASK = RAD50('WSTCAL')
00507          ,  .  (IOTYP .EQ. 2) TASK = RAD50('WSTLPR')
00508          ,  .  (IOTYP .EQ. 3) TASK = RAD50('WSTGLD')
00509          ,  .  (IOTYP .EQ. 4) TASK = RAD50('WSTTGP')
00510          ,  ...FIN
00511          ,  CALL VSNDRR (TASK,,INODE,,255,,,IDS)
00512          ,  IF (IDS .GT. 0) RETURN
00513          ,  IEGO = 3
00514          ,  RETURN
00515          ...FIN
00516          IEGO = 2
00517          RETURN
00518          END

         (FLECS VERSION 22,46)

         ----------------------------------------

00519               SUBROUTINE WLGRTV(IEGO,IDS)
00520    C ***************  [351,100]WLGRTV.FLX  ****************
00521    C
00522    C
00523    C  PROGRAM ID: CIRMIS-11-MON-SUB
00524    C
00525    C  PROGRAM DESCRIPTION:
00526    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00527    C     ROUTINE FOR ANY WELL LOG DATA REQUEST.
00528    C
00529    C
00530    C  BATTELLE MEMORIAL INSTITUTE
00531    C  PACIFIC NORTHWEST LABORATORIES
00532    C  WATER & LAND RESOURCES DEPT.
00533    C
00534    C  AUTHOR(S):   DR FRIEDRICHS
00535    C
00536    C  DATE:        INITIAL VERSION   JULY 1974
00537    C     CURRENT VERSION   APRIL 1978
00538    C
00539    C
00540               BYTE WNAME
00541    C
00542               COMMON/HDR/ INODE(256)
00543               COMMON/TND/ WNAME(12),IOTYP,IOTYP,ICTYP,NVAR,IVAR(10),KTEST
00544    C
00545               IEGO = 1
00546               IF (IOTYP .NE. 5)
00547               .  INODE(201) = IOTYP
00548               .  INODE(200) = 6
00549               .  CONDITIONAL
00550               .  .  (IOTYP .EQ. 0) TASK = RAD50('WLGSCP')
00551               .  .  (IOTYP .EQ. 1) TASK = RAD50('WLGCAL')
00552               .  .  (IOTYP .EQ. 2) TASK = RAD50('WLGLPR')
00553               .  .  (IOTYP .EQ. 3) TASK = RAD50('WLGGLD')
00554               .  .  (IOTYP .EQ. 4) TASK = RAD50('WLGTGP')
00555               .  .  (IOTYP .EQ. 6) TASK = RAD50('WLGARC')
```

```
00556          .  ...FIN
00557          .  CALL VSNDRR (TASK,,INODE,,255,,,IDS)
00558          .  IF (IDS .GT. 0) RETURN
00559          .  IEGO = 3
00560          .  RETURN
00561          ...FIN
00562          IEGO = 2
00563          RETURN
00564          END

         (FLECS VERSION 22,46)

         ------------------------------------

00565          SUBROUTINE PHYRTV(IOTYP,IEGO,IDS)
00566    C ***************  [351,100]PHYRTV.FLX  ****************
00567    C
00568    C
00569    C  PROGRAM ID: CIRMIS-11-MON-SUB
00570    C
00571    C  PROGRAM DESCRIPTION:
00572    C     THIS SUBROUTINE SETS UP THE TRANSFER NODE AND STARTS THE DESIRED
00573    C     ROUTINE FOR ANY PHYSICAL PROPERTIES DATA REQUEST.
00574    C
00575    C
00576    C  BATTELLE MEMORIAL INSTITUTE
00577    C  PACIFIC NORTHWEST LABORATORIES
00578    C  WATER & LAND RESOURCES DEPT.
00579    C
00580    C  AUTHOR(S):   DR FRIEDRICHS
00581    C     DD HOSTETLER
00582    C
00583    C  DATE:         INITIAL VERSION   SEPT. 1974
00584    C     CURRENT VERSION   SEPT. 1975
00585    C
00586    C
00587          COMMON/HDR/ INODE(256)
00588    C
00589          IEGO = 1
00590    5     READ(7,701,END=10) IDUM
00591    701   FORMAT(1115)
00592          GO TO 5
00593    10    IF (IOTYP .LE. 2)
00594          .  INODE(200) = 6
00595          .  CONDITIONAL
00596          .  .  (IOTYP .EQ. 0) TASK = RAD50('PHYSCP')
00597          .  .  (IOTYP .EQ. 1) TASK = RAD50('PHYCAL')
00598          .  .  (IOTYP .EQ. 2) TASK = RAD50('PHYLPR')
00599          .  ...FIN
00600          .  CALL REQUES(TASK,,IDS)
00601          .  IF (IDS .LE. 0) GO TO 12
00602          .  WRITE(2) TASK,INODE
00603          .  RETURN
00604          ...FIN
00605          IEGO = 2
00606          RETURN
```

```
(FLECS VERSION 22,46)      11-SEP-79   14:36:05  PAGE  0013

 00607    12     IEGO = 3
 00608           RETURN
 00609           END

          (FLECS VERSION 22,46)

          ----------------------------------------

 00610           SUBROUTINE ASTSUB (ASTCOD,LUN,IBYTES,ARRAY)
 00611           INTEGER ASTCOD,ARRAY(4)
 00612           TASK = RAD50('MNTR45')
 00613           IF(ASTCOD.NE.307) STOP 'UNEXPECTED INTERRUPT (MNTR45)'
 00614           CALL RESUME (TASK,)
 00615           END

          (FLECS VERSION 22,46)

          ----------------------------------------
```

HYDCAL.FLX

HYDROGRAPH (CALCOMP)

```
00001    C ***************  [351,100]HYDCAL.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: SRV-11-WEL-0-1       ABREVIATED WELL DESIG. VERSION
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C       THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL HYDROGRAPH DATA FROM T
00008    C       CIRMIS DATA BANK AND DRAW A CAL-COMP PLOT OF THE DATA.  THIS PROG
00009    C       IS STARTED BY 'MNTR11' AND RECEIVES CONTROL DATA VIA THE TRANSFER
00010    C       NODE.
00011    C
00012    C  DATA FILES:
00013    C       NAME                    LUN             TYPE            ACCESS
00014    C       FILE Q---FRMTHYD         1               RAN              R
00015    C       FILE Q---FRMTVTT         1               RAN              R
00016    C
00017    C
00018    C  LOADING SEQUENCE:
00019    C
00020    C       MCR>TKB @DP1:[106,215]HYDCAL.CMD
00021    C
00022    C
00023    C  BATTELLE MEMORIAL INSTITUTE
00024    C  PACIFIC NORTHWEST LABORATORIES
00025    C  WATER & LAND RESOURCES DEPT.
00026    C
00027    C  AUTHOR(S):   DR FRIEDRICHS
00028    C               SE WISE
00029    C
00030    C  DATE:        INITIAL VERSION    MAY     1974
00031    C               CURRENT VERSION    OCTOBER 1978
00032    C
00033    C
00034    C
00035          BYTE STAT,WLDES
00036          LOGICAL VTTSW
00037          DIMENSION H(512),T(512),IBUF(253),HV(256),TV(256)
00038          DIMENSION INODE(255)
00039    C
00040          COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
00041         1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
00042         2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
00043    C
00044          EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
00045          EQUIVALENCE (IDKV,IBUF(55)),(NPTV,IBUF(56)),(IDTYP,IBUF(229))
00046    C
00047    C
00048    C *** READ THE TRANSFER NODE ***
00049    C
00050    C       TASK  =  RAD50('MNTR45')
00051          CALL VRECSP (TASK,INODE,255,,,IDS)
00052          JPNT  =  0
00053          NDEL  =  0
```

```
00054   C
00055         CALL ASNLUN(1,IDISC,IUNIT)
00056         CALL DPFILE(1,'FRHTHYD',4000.,IPNT1)
00057         CALL DPFILE(1,'FRHTVTT',400.,IPNT2)
00058   C
00059   C *** RETRIEVE THE HYDROGRAPH DATA ***
00060   C
00061   C
00062   C   IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
00063   C   NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
00064   C   IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
00065   C   NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
00066   C   INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
00067   C
00068         IDKIN = IBUF(30)
00069         NPTSIN = IBUF(31)
00070         IDKEX = IBUF(32)
00071         NPTSEX = IBUF(33)
00072         INALOC = IBUF(34)
00073   C
00074   C   READ THE DATA IN THE INITIAL ALLOCATION
00075         HADSI = IDKIN
00076         WRDI = NPTSIN*2.
00077         TADSI = HADSI+INALOC
00078         CALL DPR(IPNT1,HADSI,H,WRDI)
00079         CALL DPR(IPNT1,TADSI,T,WRDI)
00080   C
00081   C   READ THE EXTENDED ALLOCATION (IF ANY)
00082   C
00083         IF (NPTSEX .NE. 0)
00084         .  HADSE = IDKEX
00085         .  WRDE  = NPTSEX*2.
00086   C     .  .
00087   C   FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
00088   C   HEIGHT AND TIME
00089   C     .
00090         .  TADSE = HADSE+2.
00091         .  NDX   = NPTSIN+1
00092         .  CALL DPR(IPNT1,HADSE,H(NDX),WRDE)
00093         .  CALL DPR(IPNT1,TADSE,T(NDX),WRDE)
00094         ...FIN
00095         NTOT = NPTSIN + NPTSEX
00096   C *** RETRIEVE VTT DATA IF REQUIRED ***
00097         VTTSW = .FALSE.
00098         IF (IDTYP. EQ .8)
00099         .  VTTSW = .TRUE.
00100         .  STRT  = IDKV
00101         .  CALL DPR(IPNT2,STRT,HV,512.)
00102         .  STRT  = IDKV+2
00103         .  CALL DPR(IPNT2,STRT,TV,512.)
00104         ...FIN
00105   C
00106         IF (MINX .EQ. 0) NEX = 0
00107         CONDITIONAL
00108         .  (NEX .EQ. 0)
00109   C     .  .
```

```
(FLECS VERSION 22.46)     11-SEP-79   14136118  PAGE  00003

00110   C *** CALCULATE AUTO SCALE FACTORS ***
00111   C
00112           .  .  TMAX = 0.0
00113           .  .  TMIN = 999999.
00114           .  .  HMAX = 0.0
00115           .  .  HMIN = 1.0E+20
00116           .  .  DO (I=1,NTOT)
00117           .  .  .  IF (T(I) .GT. TMAX) TMAX = T(I)
00118           .  .  .  IF (T(I) .LT. TMIN) TMIN = T(I)
00119           .  .  .  IF (H(I) .GT. 0.)
00120           .  .  .  .  IF (H(I) .GT. HMAX) HMAX = H(I)
00121           .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00122           .  .  .  ...FIN
00123           .  .  ...FIN
00124           .  .  MINX = IFIX(TMIN/10000.)
00125           .  .  MAXX = IFIX(TMAX/10000.)
00126           .  .  FIND=MINY=MAXY
00127           .  ...FIN
00128           .  (NEX .NE. 99)
00129           .  .  MINX = IBUF(232)
00130           .  .  MINY = IBUF(233)
00131           .  .  MAXX = IBUF(234)
00132           .  .  MAXY = IBUF(235)
00133           .  ...FIN
00134           .  (NEX .EQ. 99)
00135           .  .  MINX = IBUF(232)
00136           .  .  MAXX = IBUF(234)
00137           .  .  HMAX = 0.0
00138           .  .  HMIN = 1.0E+20
00139           .  .  DO (I=1,NTOT)
00140           .  .  .  NYR = T(I)/10000.
00141           .  .  .  IF (NYR .GE. IBUF(232) .AND. NYR .LE. IBUF(234))
00142           .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00143           .  .  .  .  IF( H(I) .GT. HMAX) HMAX = H(I)
00144           .  .  .  ...FIN
00145           .  .  ...FIN
00146           .  .  IF (HMAX .EQ. 0.) GO TO 46
00147           .  .  FIND=MINY=MAXY
00148           .  ...FIN
00149           ...FIN
00150           IMN  = 120
00151           IMX  = 700
00152           JMN  = 200
00153           JMX  = 750
00154           SCLX = IMX-IMN
00155           SCLY = JMX-JMN
00156   C
00157   C *** DRAW THE PLOT ***
00158   C
00159           CALL PLOTS(0,,0,,7)
00160           CALL NEWPEN(1)
00161           CALL FEUCL(0,0,0,0,0)
00162           CALL PLOT(0,0,0,0,-3)
00163           CALL FACTOR(1.0)
00164           SIZ  = 0.15
00165           SIZ1 = 0.1
```

```
(FLEC8 VERSION 22.46)     11-SEP-79   14:36:18  PAGE  00004

 00166          SIZ2 = 0.07
 00167    C---DEFINE BACKGROUND AND PLOT IT
 00168          CALL NEWPEN(1)
 00169          X1 = FLOAT(IMN)/100.
 00170          X2 = FLOAT(IMX)/100.
 00171          Y1 = FLOAT(JMN)/100.
 00172          Y2 = FLOAT(JMX)/100.
 00173          CALL PLOT(X1,Y1,+3)
 00174          CALL PLOT(X1,Y2,2)
 00175          CALL PLOT(X2,Y2,2)
 00176          CALL PLOT(X2,Y1,2)
 00177          CALL PLOT(X1,Y1,2)
 00178          CALL NEWPEN(1)
 00179          X = 0.7
 00180          Y = 3.2
 00181          CALL SYMBOL(X,Y,SIZ,'WATER ELEVATION (FT-MSL)',90.,24)
 00182          CALL SYMBOL(3.4,1.5,SIZ,'CALENDAR YEAR',0.,13)
 00183          CALL SYMBOL(2.95,1.0,SIZ1,'WATER LEVEL HISTORY ---',0.,23)
 00184          CALL SYMBOL(5.0,1.0,SIZ1,WLDES,0.0,12)
 00185          CALL SYMBOL(3.25,0.8,SIZ2,'CASING ELEVATION (FT) -',0.,23)
 00186          CALL NUMBER(4.75,0.8,SIZ2,CASELV,0.0,2)
 00187          NDIF = MAXY-MINY
 00188          YSCALE = SCLY/FLOAT(NDIF)
 00189    C
 00190    C----DRAW Y LINES FOR CHART
 00191          CALL NEWPEN(1)
 00192          NN = MINY
 00193          JDD = 1
 00194          NDIF = NDIF/INCY+1
 00195          DO 3604 JD = 1,NDIF
 00196          Y = (FLOAT(NN-MINY)*YSCALE+FLOAT(JMN))/100.
 00197          IF(JDD.NE.1) GO TO 3602
 00198          IF(JD.EQ.1.OR.JD.EQ.NDIF) GO TO 3601
 00199          CALL PLOT(X1,Y,3)
 00200          CALL PLOT(X2,Y,2)
 00201    3601  YY = FLOAT(NN)
 00202          CALL NUMBER(X1-.3,Y,SIZ2,YY,0.,-1)
 00203          GO TO 3603
 00204    3602  CALL PLOT(X1,Y,3)
 00205          CALL PLOT(X1+.1,Y,2)
 00206    3603  JDD = JDD+1
 00207          IF(JDD.EQ.6) JDD = 1
 00208          NN = NN+INCY
 00209    3604  CONTINUE
 00210    C
 00211    C----DRAW X LINES FOR CHART
 00212    3610  NDIF   = (MAXX-MINX) + 1
 00213          XSCALE = SCLX/(FLOAT(NDIF)*365.)
 00214          XINC   = (SCLX/FLOAT(NDIF))/100.
 00215          XADJ   = XINC/2. + SIZ2*.86
 00216          JDD = 1
 00217          DO (JD=MINX,MAXX)
 00218          .  X   = FLOAT(JD-MINX+1)*XINC + X1
 00219          .  XV  = JD
 00220          .  CALL NUMBER(X-XADJ,Y1-.16,SIZ2,XV,0.0,-1)
 00221          .  IF (JD .NE. MAXX)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:36:18  PAGE  00005

00222          .  .  SELECT(JDD)
00223          .  .  .  (1)
00224          .  .  .  .  JDD = 2
00225          .  .  .  .  CALL PLOT(X,Y1,3)
00226          .  .  .  .  CALL PLOT(X,Y1+.133,2)
00227          .  .  .  ...FIN
00228          .  .  .  (2)
00229          .  .  .  .  JDD = 1
00230          .  .  .  .  CALL PLOT(X,Y1,3)
00231          .  .  .  .  CALL PLOT(X,Y2,2)
00232          .  .  .  ...FIN
00233          .  .  ...FIN
00234          .  ...FIN
00235          ...FIN
00236    C
00237    C
00238    C---DRAW THE HYDROGRAPH
00239          NABSV = 0
00240          CALL NEWPEN(1)
00241          NPASS = 0
00242    3701  NPASS = NPASS+1
00243          IF(NPASS.EQ.1) NPTS = NTOT
00244          IF(NPASS.EQ.2) NPTS = NPTV
00245          IF(NPASS.EQ.2) CALL NEWPEN(1)
00246          NABSV = 0
00247          DO (JK = 1,NPTS)
00248          .  IF (H(JK) .GT. 0.0)
00249          .  .  WHEN (NPASS.NE.2)
00250          .  .  .  DATE = T(JK)
00251          .  .  .  ELV  = H(JK)
00252          .  .  ...FIN
00253          .  .  ELSE
00254          .  .  .  DATE = TV(JK)-19000000.
00255          .  .  .  ELV  = HV(JK)
00256          .  .  ...FIN
00257          .  .  Y    = ELV
00258          .  .  IYR  = DATE/10000.
00259          .  .  MO   = DATE/100.-FLOAT(IYR)*100.
00260          .  .  IDA  = DATE-FLOAT(MO)*100.-FLOAT(IYR)*10000.
00261    38    .  .  DAYS = (FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
00262          .  .  X    = (DAYS*XSCALE+FLOAT(IMN))/100.
00263          .  .  Y    = ((Y-FLOAT(MINY))*YSCALE+FLOAT(JMN))/100.
00264    C     .  .  CHECK FOR POINT WITHIN REGION
00265          .  .  IF(X.GE.X1. AND .X.LE.X2)
00266          .  .  .  WHEN (Y.LT.Y1. OR .Y.GT.Y2) NABS=0
00267          .  .  .  ELSE
00268    C   IF LAST VALUE OUTSIDE REGION DRAW BLANK VECTOR
00269    39    .  .  .  .  IF(.NOT.VTTSW) GO TO 3950
00270          .  .  .  .  IF(NPASS.EQ.2) GO TO 3950
00271          .  .  .  .  CALL SYMBOL(X,Y,0.07,1,0.,-1)
00272          .  .  .  .  GO TO 40
00273    3950  .  .  .  .  IF(NABSV.EQ.0) CALL PLOT(X,Y,3)
00274          .  .  .  .  IF(NABSV.EQ.1) CALL PLOT(X,Y,2)
00275          .  .  .  .  NABSV = 1
00276    40    .  .  .  .  CONTINUE
00277          .  .  .  .  IF(NPASS.EQ.2) GO TO 41
```

```
00278          . . . . IF(VTTBW) GO TO 3701
00279          . . . ...FIN
00280          . . ...FIN
00281          . ...FIN
00282          ...FIN
00283    C
00284    C
00285    41    CALL FACTOR(1.0)
00286          CALL NEWPEN(1)
00287          CALL MAPPLT (XXC,YYC,WLDES,H,T,MAPTYP)
00288          CALL PLOT(0.0,11.0,-3)
00289    45    CALL PLOTND
00290    46    IF (ICNTRL. EQ .1)
00291          . IEFN = 33
00292          . CALL SETEF(IEFN,IOS)
00293          ...FIN
00294          CALL EXIT

          ----------------------------------------

00295          TO FIND-MINY-MAXY
00296          . MINY = IFIX(HMIN/10.)*10-10
00297          . MAXY = IFIX(HMAX/10.)*10+10
00298          . INCY = 10
00299          . IF ((MAXY-MINY). LE .100) INCY = 5
00300          . IF ((MAXY-MINY). LE .40) INCY = 2
00301          ...FIN
00302          END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00295 FIND-MINY-MAXY
                00126  00147


          (FLECS VERSION 22.46)

          ----------------------------------------
```

HYDLPR.FLX

HYDROGRAPH (LINE PRINTER)

```
      -----------------------------------------
C ***************  DR0:[351,100]HYDLPR.FLX  ***************
C
C
C  PROGRAM ID: SRV-11-WEL-0-2  ABREVIATED WELL DESIG. VERSION
C
C  PROGRAM DESCRIPTION:
C      THIS PROGRAM IS DESIGNED TO RETRIEVE WELL HYDROGRAPH DATA FROM TH
C      CIRMIS DATA BANK AND PRINT THE DATA IN FORMATED FORM ON THE LINE
C      PRINTER.  THIS PROGRAM IS STARTED BY 'MNTR11' AND RECEIVES CONTRO
C      DATA VIA THE TRANSFER NODE DP1:[22,33]TNODE.RAN.
C
C  DATA FILES:
C      NAME                  LUN              TYPE              ACCESS
C      FILE 0--FRMTHYD        1               RAN                 R
C
C
C  LOADING SEQUENCE:
C
C      MCR>TKB @DB0:[351,100]HYDLPR
C
C
C  BATTELLE MEMORIAL INSTITUTE
C  PACIFIC NORTHWEST LABORATORIES
C  WATER & LAND RESOURCES DEPT.
C
C  AUTHOR(S):    DR FRIEDRICHS
C                SE WISE
C            MODIFICATIONS FOR TERMINAL OUTPUT BY RC ARNETT, ARHCO
C
C
C  DATE:         INITIAL VERSION    MAY       1974
C                CURRENT VERSION    OCTOBER   1976
C
C
C
      BYTE DT(512,9),AMON(3,12),YR(2),DAY(2),STAT,WLDES
      DIMENSION H(512),T(512),IBUF(255)
      DIMENSION INODE(257)
C
      COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
     1          ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
     2          NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
C
      EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
C
      DATA AMON/'J','A','N','F','E','B','M','A','R','A','P','R',
     1 'M','A','Y','J','U','N','J','U','L','A','U','G','S','E','P',
     2 'O','C','T','N','O','V','D','E','C'/
C
C
C *** READ THE TRANSFER NODE ***
C
C     TASK  =  RAD50('MNTR45')
```

```
(FLECS VERSION 22.46)     11-SEP-79   14137107  PAGE  00002

 00054           CALL VRECSP (TASK,INODE,255,,,IDS)
 00055    C      IBUF(200) IS A FLAG, 5 = TI1 TERMINAL, ANYTHING ELSE
 00056    C      WILL SEND OUTPUT TO THE LINEPRINTER, DESIGNED FOR
 00057    C      ACCESS BY ARHCO TERMINAL
 00058    C
 00059           IDISC = IBUF(251)
 00060           IUNIT = IBUF(252)
 00061           CALL ASNLUN(1,IDISC,IUNIT)
 00062           CALL OPFILE(1,'FRMTHYD',4000,,IPNT1)
 00063    C
 00064           LUN1 = 6
 00065           IF(IBUF(200).EQ.5)LUN1 = 5
 00066    C
 00067    C *** RETRIEVE DATA ***
 00068    C
 00069    C    NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
 00070    C    IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
 00071    C
 00072           IDKIN  = IBUF(30)
 00073           NPTSIN = IBUF(31)
 00074    C
 00075    C    NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
 00076    C    IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
 00077    C
 00078           IDKEX  = IBUF(32)
 00079           NPTSEX = IBUF(33)
 00080    C
 00081    C    INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
 00082    C
 00083           INALOC = IBUF(34)
 00084    C
 00085    C    READ THE DATA IN THE INITIAL ALLOCATION
 00086    C
 00087           HADSI = IDKIN
 00088           WRDI  = NPTSIN*2.
 00089           TADSI = HADSI+INALOC
 00090           CALL DPR(IPNT1,HADSI,H,WRDI)
 00091           CALL DPR(IPNT1,TADSI,T,WRDI)
 00092    C
 00093    C    READ THE EXTENDED ALLOCATION (IF ANY)
 00094    C
 00095           IF (NPSEX .NE. 0)
 00096    C      .
 00097    C    FOUR SECTORS ARE ALLOCATED IN THE EXTENSION, TWO EACH
 00098    C    FOR HEIGHT AND TIME
 00099    C      .
 00100           .  HADSE = IDKEX
 00101           .  WRDE  = NPTSEX*2.
 00102           .  TADSE = HADSE+2.
 00103           .  NDX   = NPTSIN+1
 00104           .  CALL DPR(IPNT1,HADSE,H(NDX),WRDE)
 00105           .  CALL DPR(IPNT1,TADSE,T(NDX),WRDE)
 00106           ...FIN
 00107           CALL DPCL (IPNT1)
 00108           NPTS = NPTSIN + NPTSEX
 00109    C
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:37:07  PAGE  00003

  00110    C *** WRITE HEADER INFORMATION TO THE PRINTER ***
  00111    C
  00112          WRITE(LUN1,500) WLDES,NPTS,CASELV
  00113    500   FORMAT(22X,'HYDROGRAPH DATA - WELL NO. ',12A1//
  00114         1 26X,'MEASURMENTS TO DATE = ',I3/
  00115         2 26X,'CASING ELEVATION    = ',F6.2/
  00116         3 26X,'      *******       = DRY WELL'//)
  00117    C
  00118    C *** CONVERT THE DATE TO ASCII ***
  00119    C
  00120          DO (I = 1,NPTS)
  00121          .  DT(I,3) = '-'
  00122          .  DT(I,7) = '-'
  00123          .  IYR =  T(I)/10000.
  00124          .  IMON = T(I)/100.-IYR*100.
  00125          .  WHEN (IMON .GT. 0)
  00126          .  .  IDAY    = T(I)-IMON*100.-IYR*10000.
  00127          .  .  DT(I,4) = AMON(1,IMON)
  00128          .  .  DT(I,5) = AMON(2,IMON)
  00129          .  .  DT(I,6) = AMON(3,IMON)
  00130          .  .  ENCODE (2,501,YR) IYR
  00131    501   .  .  FORMAT(I2)
  00132          .  .  ENCODE (2,501,DAY) IDAY
  00133          .  .  DT(I,1) = DAY(1)
  00134          .  .  DT(I,2) = DAY(2)
  00135          .  .  DT(I,8) = YR(1)
  00136          .  .  DT(I,9) = YR(2)
  00137          .  ...FIN
  00138          .  ELSE
  00139          .  .  DO (M=1,9) DT(I,M)=' '
  00140          .  ...FIN
  00141          ...FIN
  00142    C
  00143    C *** DUMP THE DATA TO THE PRINTER ***
  00144    C
  00145          WRITE(LUN1,502) ((DT(K,M),M=1,9),H(K),K=1,NPTS)
  00146    502   FORMAT(3(5X,9A1,2X,F7.2,3X))
  00147          IF (ICNTRL. EQ .1)
  00148          .  IEFN = 33
  00149          .  CALL SETEF(IEFN,IDS)
  00150          ...FIN
  00151          CALL EXIT
  00152          END

         (FLECS VERSION 22.46)

         ----------------------------------------
```

HYDGLD.FLX

HYDROGRAPH (GOULD)

```
00001    C ***************  [351,100]HYDGLD.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: SRV-11-WEL-1-3
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL HYDROGRAPH DATA FROM THE
00008    C     CIRMIS DATA BANK AND DRAW A GOULD PLOT OF THE DATA.  THIS PROGRAM
00009    C     IS STARTED BY 'MNTR11' AND RECEIVES CONTROL DATA VIA THE TRANSFER
00010    C     NODE.
00011    C
00012    C  DATA FILES:
00013    C        NAME                        LUN              TYPE             ACCESS
00014    C        FILE 0---FRMTHYD             1               RAN               R
00015    C
00016    C
00017    C  LOADING SEQUENCE:
00018    C
00019    C     MCR>TKB @[351,100]HYDGLD
00020    C
00021    C
00022    C  BATTELLE MEMORIAL INSTITUTE
00023    C  PACIFIC NORTHWEST LABORATORIES
00024    C  WATER & LAND RESOURCES DEPT.
00025    C
00026    C  AUTHOR(S):    DR FRIEDRICHS
00027    C
00028    C  DATE:          INITIAL VERSION       JANUARY 1977
00029    C                 CURRENT VERSION       JUNE    1978
00030    C
00031    C
00032          BYTE A(3),B(2),WLDES
00033    C
00034          INTEGER BUFFER,SIZ,SIZ1,SIZ2,YC
00035          DIMENSION H(512),T(512),INODE(255),IBUF(253)
00036    C
00037          COMMON/BFR/ BUFFER(4800)
00038    C
00039          COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
00040         1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
00041         2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
00042    C
00043          EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
00044          EQUIVALENCE (IDKV,IBUF(55)),(NPTV,IBUF(56)),(IDTYP,IBUF(229))
00045    C
00046    C
00047    C *** READ THE TRANSFER NODE ***
00048    C
00049    C     TASK  =  RAD50('MNTR45')
00050          CALL VRECSP (TASK,INODE,255,,,IOS)
00051          JPNT  =  0
00052          NDEL  =  0
00053    C
```

```
00054          CALL ASNLUN(1,IDISC,IUNIT)
00055          CALL OPFILE(1,'FRMTHYD',4000.,IPNT1)
00056          CALL OPFILE(1,'FRMTVTT',400.,IPNT2)
00057    C
00058    C *** RETRIEVE THE HYDROGRAPH DATA ***
00059    C
00060    C
00061    C    IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
00062    C    NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
00063    C    IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
00064    C    NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
00065    C    INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
00066    C
00067          IDKIN = IBUF(30)
00068          NPTSIN = IBUF(31)
00069          IDKEX = IBUF(32)
00070          NPTSEX = IBUF(33)
00071          INALOC = IBUF(34)
00072    C
00073    C    READ THE DATA IN THE INITIAL ALLOCATION
00074          HADSI = IDKIN
00075          WRDI = NPTSIN*2.
00076          TADSI = HADSI+INALOC
00077          CALL DPR(IPNT1,HADSI,H,WRDI)
00078          CALL DPR(IPNT1,TADSI,T,WRDI)
00079    C
00080    C    READ THE EXTENDED ALLOCATION (IF ANY)
00081    C
00082          IF (NPTSEX .NE. 0)
00083         .  HADSE = IDKEX
00084         .  WRDE  = NPTSEX*2.
00085    C    .
00086    C    FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
00087    C    HEIGHT AND TIME
00088    C    .
00089         .  TADSE = HADSE+2.
00090         .  NDX   = NPTSIN+1
00091         .  CALL DPR(IPNT1,HADSE,H(NDX),WRDE)
00092         .  CALL DPR(IPNT1,TADSE,T(NDX),WRDE)
00093         ...FIN
00094          NTOT = NPTSIN + NPTSEX
00095    C *** RETRIEVE VTT DATA IF REQUIRED ***
00096          VTTSW = .FALSE.
00097          IF (IDTYP. EQ .8)
00098         .  VTTSW = .TRUE.
00099         .  STRT  = IDKV
00100         .  CALL DPR(IPNT2,STRT,HV,512.)
00101         .  STRT  = IDKV+2
00102         .  CALL DPR(IPNT2,STRT,TV,512.)
00103         ...FIN
00104    C
00105          IF (MINX .EQ. 0) NEX = 0
00106          CONDITIONAL
00107         .  (NEX .EQ. 0)
00108    C    .  .
00109    C *** CALCULATE AUTO SCALE FACTORS ***
```

```
00110   C       .  .
00111           .  .  TMAX = 0.0
00112           .  .  TMIN = 999999.
00113           .  .  HMAX = 0.0
00114           .  .  HMIN = 1.0E+20
00115           .  .  DO (I=1,NTOT)
00116           .  .  .  IF (T(I) .GT. TMAX) TMAX = T(I)
00117           .  .  .  IF (T(I) .LT. TMIN) TMIN = T(I)
00118           .  .  .  IF (H(I) .GT. 0.)
00119           .  .  .  .  IF (H(I) .GT. HMAX) HMAX = H(I)
00120           .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00121           .  .  .  ...FIN
00122           .  .  ...FIN
00123           .  .  MINX = IFIX(TMIN/10000.)
00124           .  .  MAXX = IFIX(TMAX/10000.)
00125           .  .  FIND=MINY-MAXY
00126           .  ...FIN
00127           .  (NEX .NE. 99)
00128           .  .  MINX = IBUF(232)
00129           .  .  MINY = IBUF(233)
00130           .  .  MAXX = IBUF(234)
00131           .  .  MAXY = IBUF(235)
00132           .  ...FIN
00133           .  (NEX .EQ. 99)
00134           .  .  MINX = IBUF(232)
00135           .  .  MAXX = IBUF(234)
00136           .  .  HMAX = 0.0
00137           .  .  HMIN = 1.0E+20
00138           .  .  DO (I=1,NTOT)
00139           .  .  .  NYR = T(I)/10000.
00140           .  .  .  IF (NYR .GE. IBUF(232) .AND. NYR .LE. IBUF(234))
00141           .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00142           .  .  .  .  IF( H(I) .GT. HMAX) HMAX = H(I)
00143           .  .  .  ...FIN
00144           .  .  ...FIN
00145           .  .  IF (HMAX .EQ. 0.) GO TO 46
00146           .  .  FIND=MINY-MAXY
00147           .  ...FIN
00148           ...FIN
00149   14      IMN=170
00150           IMX=770
00151           JMN=300
00152           JMX=900
00153           SCLX=IMX-IMN
00154           SCLY=JMX-JMN
00155           SIZ=24
00156           SIZ1=16
00157           SIZ2=8
00158   C---DEFINE BACKGROUND AND PLOT IT
00159           X1=FLOAT(IMN)
00160           X2=FLOAT(IMX)
00161           Y1=FLOAT(JMN)
00162           Y2=FLOAT(JMX)
00163   C
00164   C *** DRAW THE PLOT ***
00165   C
```

```
(FLECS VERSION 22.46)     11-SEP-79   14138151  PAGE  00004

  00166          CALL DOTINI(7,BUFFER,32,1200)
  00167          DO 9990 INDEX=1,864,32
  00168    C
  00169          CALL FEDCL(0.0,0.0,1)
  00170    C---DEFINE BACKGROUND AND PLOT IT
  00171          CALL DOTLIN(IMN,JMN,IMN,JMX)
  00172          CALL DOTLIN(IMN,JMX,IMX,JMX)
  00173          CALL DOTLIN(IMX,JMX,IMX,JMN)
  00174          CALL DOTLIN(IMX,JMN,IMN,JMN)
  00175          IXS=120
  00176          IYS=390
  00177          CALL DOTCHR(IXS,IYS,'WATER ELEVATION (FT-MSL)',,SIZ,1)
  00178          CALL DOTCHR(360,250,'CALENDAR YEAR',,SIZ)
  00179          IX=360
  00180          IY=150
  00181          IZ=SIZ1
  00182          CALL DOTCHR(IX,IY,'WATER LEVEL HISTORY',,IZ)
  00183          IXL=(SCLX/2.) - FLOAT(SIZ2)*.75*16. + X1
  00184          IY=130
  00185          CALL DOTCHR(IXL,IY,'WELL DESIGNATION = ',,SIZ2)
  00186          CALL DOTCHR(IXL+120,IY,WLDES,12,SIZ2)
  00187          NDIF=MAXY-MINY
  00188          YSCALE=SCLY/FLOAT(NDIF)
  00189    C
  00190    C
  00191    C----DRAW Y LINES FOR CHART
  00192          NN=MINY
  00193          JDD=1
  00194          NDIF=NDIF/INCY+1
  00195          DO 3604 JD=1,NDIF
  00196          Y= FLOAT(NN-MINY)*YSCALE+FLOAT(JMN)
  00197          IY = Y
  00198          IF(JDD.NE.1) GO TO 3602
  00199          IF(JD.EQ.1.OR.JD.EQ.NDIF) GO TO 3601
  00200          CALL DOTLIN (IMN,IY,IMX,IY)
  00201    3601  ENCODE (3,321,A) NN
  00202    321   FORMAT(I3)
  00203          CALL DOTCHR(IMN-30,IY,A,3,,ISIZ2)
  00204          GO TO 3603
  00205    3602  CALL DOTLIN (IMN,IY,IMN+10,IY)
  00206    3603  JDD=JDD+1
  00207          IF(JDD.EQ.6) JDD=1
  00208          NN=NN+INCY
  00209    3604  CONTINUE
  00210    C
  00211    C
  00212    C----DRAW X LINES FOR CHART
  00213          NDIF=(MAXX-MINX) + 1
  00214          XSCALE=SCLX/(FLOAT(NDIF)*365.)
  00215          XINC=(SCLX/FLOAT(NDIF))
  00216          XADJ=XINC/2. + FLOAT(SIZ2)*.75
  00217          JDD=1
  00218          DO 3650 JD=MINX,MAXX
  00219          X=FLOAT(JD-MINX+1)*XINC + X1
  00220          IX=X-XADJ
  00221          ENCODE(2,636,8) JD
```

```
00222   636   FORMAT(I2)
00223         CALL DOTCHR(IX,JMN-20,8,2,8IZ2)
00224         IX=X
00225         IF(JD.EQ.MAXX) GO TO 3650
00226         GO TO(3620,3630),JDD
00227   3620  JDD=2
00228         CALL DOTLIN(IX,JMN,IX,JMN+10)
00229         GO TO 3650
00230   3630  JDD=1
00231         CALL DOTLIN(IX,JMN,IX,JMX)
00232   3650  CONTINUE
00233   C
00234   C
00235   C---DRAW THE GRAPH
00236   C
00237         IF(INDEX.EQ.1) WRITE(6,637) NTOT,(T(I),H(I),I=1,NTOT)
00238   637   FORMAT(' NTOT= ',I3,/,(1X,2F10.2))
00239         NARSV=0
00240         DO 40 JK=1,NTOT
00241   3701  Y=H(JK)
00242         IYR=T(JK)/10000.
00243         MO=T(JK)/100.-FLOAT(IYR)*100.
00244         IDA=T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
00245   38    DAYS=(FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
00246         X=(DAYS*XSCALE+FLOAT(IMN))
00247         Y = (Y-FLOAT(MINY))*YSCALE + JMN
00248         IX = X
00249         IY = Y
00250   C     CHECK FOR POINT WITHIN REGION
00251   3806  IF(X.LT.X1.OR.X.GT.X2) GO TO 3850
00252         IF(Y.GE.Y1.AND.Y.LE.Y2) GO TO 39
00253   3850  NARSV = 0
00254         GO TO 3990
00255   C  YES---PLOT DATA POINT
00256   39    IF (NARSV.EQ.0) GO TO 3980
00257         CALL DOTLIN (LASTX,LASTY,IX,IY)
00258   3980  NARSV = 1
00259   3990  LASTX = IX
00260         LASTY = IY
00261   40    CONTINUE
00262   C
00263   C  PLOT DATA POINTS
00264   9990  CALL DOTOUT
00265         CALL DOTDUN
00266   46    IF (ICNTRL. EQ .1)
00267         .  IEFN = 33
00268         .  CALL SETEF(IEFN,IDS)
00269         ...FIN
00270         CALL EXIT

      ----------------------------------------

00271         TO FIND-MINY-MAXY
00272         .  MINY = IFIX(HMIN/10.)*10-10
00273         .  MAXY = IFIX(HMAX/10.)*10+10
00274         .  INCY = 10
```

```
00275          .  IF ((MAXY-MINY). LE .100) INCY = 5
00276          .  IF ((MAXY-MINY). LE .40) INCY = 2
00277          ...FIN
00278          END
```

PROCEDURE CROSS-REFERENCE TABLE

00271 FIND-MINY-MAXY
00125 00146

(FLECS VERSION 22.46)

```
00279    C
00280    C
00281          SUBROUTINE FEDCL(XX,YY,IR)
00282    C
00283          INTEGER X,Y,BUFFER
00284    C
00285          COMMON/BFR/ BUFFER(4800)
00286    C
00287    C
00288          XMAX=850.
00289          YMAX=1100.
00290          IF(IR.EQ.1) GO TO 10
00291          XMAX=1100.
00292          YMAX=850.
00293    10    Y=YY
00294          DO 15 J=1,4
00295          X=XX
00296          IF(J.GE.3) Y=YY+YMAX
00297          IF(J.EQ.2.OR.J.EQ.4) X=XX+XMAX
00298          CALL DOTLIN(X,Y+10,X+20,Y+10)
00299          CALL DOTLIN(X+10,Y,X+10,Y+20)
00300    15    CONTINUE
00301          RETURN
00302          END
```

(FLECS VERSION 22.46)

CTMCAL.FLX

CONTAMINATION (CALCOMP)

```
00001    C ***************  DR0:[351,100]CTMCAL.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: SRV-11-WEL-1-1 ABREVIATED WELL DESIG, VERSION
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL CONTAMINATE DATA FROM THE
00008    C     CIRMIS DATA BANK AND DRAW A CAL-COMP PLOT OF THE DATA,  THIS PROGRAM
00009    C     IS STARTED BY 'MNTR11' AND RECEIVES CONTROL DATA VIA THE TRANSFER
00010    C     NODE.
00011    C
00012    C  DATA FILES:
00013    C       NAME                    LUN          TYPE          ACCESS
00014    C       FILE 0---FRMTCTM         1            RAN            R
00015    C       FILE 0---CTMHDR          1            RAN            R
00016    C
00017    C
00018    C  LOADING SEQUENCE:
00019    C
00020    C     MCR>TKB @DB0:[351,100]CTMCAL
00021    C
00022    C
00023    C  BATTELLE MEMORIAL INSTITUTE
00024    C  PACIFIC NORTHWEST LABORATORIES
00025    C  WATER & LAND RESOURCES DEPT.
00026    C
00027    C  AUTHOR(S):   DR FRIEDRICHS
00028    C               SE WISE
00029    C               DW DAMSCHEN
00030    C
00031    C  DATE:        INITIAL VERSION  MAY    1974
00032    C               CURRENT VERSION AUGUST  1979
00033    C
00034    C
00035          BYTE STAT,WLDES
00036    C
00037          INTEGER*4 CTIADR(36), CTXADR(36)
00038    C
00039          DIMENSION M(512),T(512),IBUF(253),XDEL(17),YDEL(17)
00040          DIMENSION INODE(255), KBUF(256)
00041    C
00042          COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
00043         1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
00044         2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
00045    C
00046          EQUIVALENCE (WLDES(1),IBUF(1))
00047          EQUIVALENCE (IDM(1),INODE(1)), (CTIADR(1),KBUF(1))
00048          EQUIVALENCE (CTXADR(1),KBUF(73))
00049    C
00050    C *** READ THE TRANSFER NODE ***
00051    C
00052    C     TASK   =  RAD50('MNTR45')
00053          CALL VRECSP (TASK,INODE,255,,,IDS)
```

```
00054          IPNT   = 0
00055          NDEL   = 0
00056    C
00057          CALL ASNLUN(1,IDISC,IUNIT)
00058          CALL OPFILE(1,'FRMTCTM',37000,,IPNT1)
00059          CALL OPFILE(1,'CTMHDR',6000,,ICHDR)
00060    C
00061    C *** RETRIEVE THE CONTAMINATION DATA ***
00062    C
00063          QADR = IBUF(61)
00064          CALL DPR(ICHDR,QADR,KBUF,256,)
00065    C
00066          IOFF = (ICTM-1)*3
00067          NPTS = KBUF(145+IOFF)
00068          DKSAV= CTIADR(ICTM)
00069          DKSEX= CTXADR(ICTM)
00070          NPSEX= KBUF(146+IOFF)
00071          INALOC=KBUF(147+IOFF)
00072    C
00073    C
00074          WRDI = 2*NPTS
00075          WI   = INALOC*256
00076          DKT  = DKSAV+INALOC
00077          DKM  = DKT+INALOC
00078          CALL DPR(IPNT1,DKSAV,H,WRDI)
00079          CALL DPR(IPNT1,DKT,T,WRDI)
00080          IF (NPSEX .NE. 0)
00081    C      .
00082    C     READ EXTENDED AREA (IF ANY)
00083    C      .
00084          .  WRDE = 2*NPSEX
00085          .  DKTE = DKSEX+2,
00086          .  DKME = DKTE+2,
00087          .  NPX  = NPTS+1
00088          .  CALL DPR(IPNT1,DKSEX,H(NPX),WRDE)
00089          .  CALL DPR(IPNT1,DKTE,T(NPX),WRDE)
00090          ...FIN
00091    C    REMOVE ANY MINUS VALUES
00092          NTOT = NPTS+NPSEX
00093          DO (I=1,NTOT)
00094          .  H(I) = ABS(H(I))
00095          ...FIN
00096    C
00097    C
00098          WRITE (5,888) WLDES,ICTM,NEX,MINX,MAXX
00099    888   FORMAT(1X,12A1,'CTM= ',I2,' NEX= ',I3,' MINX= ',I3,
00100         1 ' MAXX= ',I3)
00101    C
00102          IF (MINX .EQ. 0) NEX = 0
00103          CONDITIONAL
00104          .  (NEX .EQ. 0)
00105    C ** DETERMINE MIN & MAX X & Y,
00106    C      . .
00107    C *** CALCULATE AUTO SCALE FACTORS ***
00108    C      . .
00109          .  .  TMAX = 0.0
```

```
00110          . .  TMIN = 999999.
00111          . .  HMAX = 0.0
00112          . .  HMIN = 1.0E+20
00113          . .  DO (I=1,NTOT)
00114          . .  .  IF (T(I) .GT. TMAX) TMAX = T(I)
00115          . .  .  IF (T(I) .LT. TMIN) TMIN = T(I)
00116          . .  .  IF (H(I) .GT. 0.)
00117          . .  .  .  IF (H(I) .GT. HMAX) HMAX = H(I)
00118          . .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00119          . .  .  ...FIN
00120          . .  ...FIN
00121          . .  MINX = IFIX(TMIN/10000.)
00122          . .  MAXX = IFIX(TMAX/10000.)
00123          . .  FIND=MINY-MAXY
00124          . ...FIN
00125          . (NEX .EQ. 99)
00126    C ** DETERMINE MIN & MAX Y FROM INPUT MIN & MAX X.
00127          . .  HMAX = 0.0
00128          . .  HMIN = 1.0E+20
00129          . .  J = 0
00130          . .  DO (I=1,NTOT)
00131          . .  .  NYR = T(I)/10000.
00132          . .  .  IF (NYR .GE. MINX .AND. NYR .LE. MAXX)
00133          . .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00134          . .  .  .  IF( H(I) .GT. HMAX) HMAX = H(I)
00135          . .  .  .  J = J + 1
00136          . .  .  .  H(J) = H(I)
00137          . .  .  .  T(J) = T(I)
00138          . .  .  ...FIN
00139          . .  ...FIN
00140          . .  NTOT = J
00141          . .  IF (NTOT .LE. 0) IDM1(1) = 0
00142          . .  IF (NTOT .LE. 0) GO TO 46
00143          . .  IF (HMAX .EQ. 0.) GO TO 46
00144          . .  FIND=MINY-MAXY
00145          . ...FIN
00146          ...FIN
00147          IMN  = 120
00148          IMX  = 700
00149          JMN  = 200
00150          JMX  = 750
00151          SCLX = IMX-IMN
00152          SCLY = JMX-JMN
00153    C
00154    C *** DRAW THE PLOT ***
00155    C
00156          CALL PLOTS(0.,0.,7)
00157          CALL NEWPEN(1)
00158          CALL FEDCL(0.0,0.0,0.0,0)
00159          CALL PLOT(0.0,0.0,-3)
00160          CALL FACTOR(1.0)
00161          SIZ  = 0.15
00162          SIZ1 = 0.1
00163          SIZ2 = 0.07
00164    C---DEFINE BACKGROUND AND PLOT IT
00165          CALL NEWPEN(1)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:11  PAGE  00004

  00166          X1 = FLOAT(IMN)/100.
  00167          X2 = FLOAT(IMX)/100.
  00168          Y1 = FLOAT(JMN)/100.
  00169          Y2 = FLOAT(JMX)/100.
  00170          CALL PLOT(X1,Y1,+3)
  00171          CALL PLOT(X1,Y2,2)
  00172          CALL PLOT(X2,Y2,2)
  00173          CALL PLOT(X2,Y1,2)
  00174          CALL PLOT(X1,Y1,2)
  00175          CALL NEWPEN(1)
  00176          XS = 0.7
  00177          YS = 3.2
  00178          CALL SYMBOL(XS,YS,SIZ,'CONCENTRATION',90.,13)
  00179          YS = YS+1.9
  00180          GO TO(15,15,15,16,15,15,15,17,15,15,16,16,16,
  00181         1      16,16,16,16,16,16,16,16,2050,16,16,16,15,
  00182         2      16,16,18,19,20,15),ICTM
  00183    15    CALL SYMBOL(XS,YS,SIZ,'(PCI/ML)',90.,8)
  00184          GO TO 2050
  00185    16    CALL SYMBOL(XS,YS,SIZ,'(MG/L)',90.,6)
  00186          GO TO 2050
  00187    17    CALL SYMBOL(XS,YS,SIZ,'(NG/ML)',90.,7)
  00188          GO TO 2050
  00189    18    CALL SYMBOL(XS,YS,SIZ,'(PPM)',90.,5)
  00190          GO TO 2050
  00191    19    CALL SYMBOL(XS,YS,SIZ,'(UMHOS/CM3)',90.,11)
  00192          GO TO 2050
  00193    20    CALL SYMBOL(XS,YS,SIZ,'(UG/L)',90.,6)
  00194    2050  CALL SYMBOL(3.4,1.5,SIZ,'CALENDAR YEAR',0.,13)
  00195          X = 2.74
  00196          Y = 1.0
  00197          Z = SIZ1
  00198          CALL SYMBOL(X,Y,Z,'CONCENTRATION HISTORY ---',0.,25)
  00199          X = 5.0
  00200          IF(ICTM.EQ.1) CALL SYMBOL(X,Y,Z,'TOTAL ALPHA',0.,11)
  00201          IF(ICTM.EQ.2) CALL SYMBOL(X,Y,Z,'TOTAL BETA',0.,10)
  00202          IF(ICTM.EQ.3) CALL SYMBOL(X,Y,Z,'TRITIUM',0.,7)
  00203          IF(ICTM.EQ.4) CALL SYMBOL(X,Y,Z,'NITRATE',0.,7)
  00204          IF(ICTM.EQ.5) CALL SYMBOL(X,Y,Z,'STRONTIUM-90',0.,12)
  00205          IF(ICTM.EQ.6) CALL SYMBOL(X,Y,Z,'CESIUM-137',0.,10)
  00206          IF(ICTM.EQ.7) CALL SYMBOL(X,Y,Z,'COBALT-60',0.,9)
  00207          IF(ICTM.EQ.8) CALL SYMBOL(X,Y,Z,'URANIUM-238',0.,11)
  00208          IF(ICTM.EQ.9) CALL SYMBOL(X,Y,Z,'PLUTONIUM-239',0.,13)
  00209          IF(ICTM.EQ.10)CALL SYMBOL(X,Y,Z,'RUTHENIUM-106',0.,13)
  00210          IF(ICTM.EQ.11)CALL SYMBOL(X,Y,Z,'CHROMIUM',0.,8)
  00211          IF(ICTM.EQ.12)CALL SYMBOL(X,Y,Z,'FLUORIDE',0.,8)
  00212          IF(ICTM.EQ.13)CALL SYMBOL(X,Y,Z,'MAGNESIUM',0.,9)
  00213          IF(ICTM.EQ.14)CALL SYMBOL(X,Y,Z,'IRON',0.,4)
  00214          IF(ICTM.EQ.15)CALL SYMBOL(X,Y,Z,'PHOSPHATE',0.,9)
  00215          IF(ICTM.EQ.16)CALL SYMBOL(X,Y,Z,'CHLORIDE',0.,8)
  00216          IF(ICTM.EQ.17)CALL SYMBOL(X,Y,Z,'COPPER',0.,6)
  00217          IF(ICTM.EQ.18)CALL SYMBOL(X,Y,Z,'HARDNESS',0.,8)
  00218          IF(ICTM.EQ.19)CALL SYMBOL(X,Y,Z,'SOLIDS',0.,6)
  00219          IF(ICTM.EQ.20)CALL SYMBOL(X,Y,Z,'MANGANESE',0.,9)
  00220          IF(ICTM.EQ.21)CALL SYMBOL(X,Y,Z,'TOTAL ORGANIC CARBON',0.,20)
  00221          IF(ICTM.EQ.22)CALL SYMBOL(X,Y,Z,'PH',0.,2)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:11  PAGE  00005

  00222          IF(ICTM.EQ.23)CALL SYMBOL(X,Y,Z,'SULFATE',0.,7)
  00223          IF(ICTM.EQ.24)CALL SYMBOL(X,Y,Z,'SODIUM',0.,6)
  00224          IF(ICTM.EQ.25)CALL SYMBOL(X,Y,Z,'CALCIUM',0.,7)
  00225          IF(ICTM.EQ.26)CALL SYMBOL(X,Y,Z,'TOTAL GAMMA',0.,11)
  00226          IF(ICTM.EQ.27)CALL SYMBOL(X,Y,Z,'BICARBONATE ION',0.,15)
  00227          IF(ICTM.EQ.28)CALL SYMBOL(X,Y,Z,'CARBONATE ION',0.,13)
  00228          IF(ICTM.EQ.29)CALL SYMBOL(X,Y,Z,'TOTAL POTASSIUM',0.,15)
  00229          IF(ICTM.EQ.30)CALL SYMBOL(X,Y,Z,'SPECIFIC CONDUCTIVITY',0.,21)
  00230          IF(ICTM.EQ.31)CALL SYMBOL(X,Y,Z,'BORON',0.,5)
  00231          IF(ICTM.EQ.32)CALL SYMBOL(X,Y,Z,'LOW ALPHA',0.,9)
  00232          XL = (SCLX/2.)/100. - SIZ2*0.86*16. + X1
  00233          Y  = 0.8
  00234          CALL NEWPEN(1)
  00235          CALL SYMBOL(XL,Y,SIZ2,'WELL DESIGNATION - ',0.0,19)
  00236          CALL SYMBOL(XL+1.14,Y,SIZ2,WLDES,0.0,12)
  00237          NDIF   = MAXY-MINY
  00238          YSCALE = SCLY/FLOAT(NDIF)
  00239          SIZ2   =  0.09
  00240     C----DRAW Y LINES FOR CHART
  00241          CALL NEWPEN(1)
  00242          IYDF = (JMX-JMN)/NDIF
  00243          IDR  = 1
  00244          DO (JD=1,NDIF)
  00245          .  YC = (JMN+(JD-1)*IYDF)/100.
  00246          .  NN = MINY+JD-1
  00247          .  IF (JD .NE. 1)
  00248          .  .  SELECT(IDR)
  00249          .  .  .  (1)
  00250          .  .  .  .  CALL PLOT(X1,YC,3)
  00251          .  .  .  .  CALL PLOT(X2,YC,2)
  00252          .  .  .  .  IDR = 2
  00253          .  .  .  ...FIN
  00254          .  .  .  (2)
  00255          .  .  .  .  CALL PLOT(X2,YC,3)
  00256          .  .  .  .  CALL PLOT(X1,YC,2)
  00257          .  .  .  .  IDR = 1
  00258          .  .  .  ...FIN
  00259          .  .  ...FIN
  00260          .  ...FIN
  00261          .  YY = FLOAT(NN)
  00262          .  CALL SYMBOL(X1-.42,YC,SIZ2,'1,E',0.,3)
  00263          .  CALL NUMBER(X1-.2,YC,SIZ2,YY,0.,-1)
  00264          .  NB = 1
  00265          .  IF (JD .EQ. 1) NB = 2
  00266          .  DO (K=NB,9)
  00267          .  .  Y = YC+(ALOG10(FLOAT(K))*IYDF)/100.
  00268          .  .  SELECT(IDR)
  00269          .  .  .  (1)
  00270          .  .  .  .  CALL PLOT(X1,Y,3)
  00271          .  .  .  .  CALL PLOT(X2,Y,2)
  00272          .  .  .  .  IDR = 2
  00273          .  .  .  ...FIN
  00274          .  .  .  (2)
  00275          .  .  .  .  CALL PLOT(X2,Y,3)
  00276          .  .  .  .  CALL PLOT(X1,Y,2)
  00277          .  .  .  .  IDR = 1
```

```
00278          .  .  .  ...FIN
00279          .  .  ...FIN
00280          .  ...FIN
00281          ...FIN
00282          CALL SYMBOL(X1-.42,Y2,SIZ2,'1.E',0.,3)
00283          YV = FLOAT(MAXY)
00284          CALL NUMBER(X1-.2,Y2,SIZ2,YV,0.,-1)
00285    C----DRAW X LINES FOR CHART
00286    3610  NDIF   = (MAXX-MINX) + 1
00287          XSCALE = SCLX/(FLOAT(NDIF)*365.)
00288          XINC   = (SCLX/FLOAT(NDIF))/100.
00289          XADJ   = XINC/2. + SIZ2*.86
00290          JDD = 1
00291          DO (JD=MINX,MAXX)
00292          .  X    = FLOAT(JD-MINX+1)*XINC + X1
00293          .  XV   = JD
00294          .  CALL NUMBER(X-XADJ,Y1-.16,SIZ2,XV,0.0,-1)
00295          .  IF (JD .NE. MAXX)
00296          .  .  SELECT(JDD)
00297          .  .  .  (1)
00298          .  .  .  .  JDD = 2
00299          .  .  .  .  CALL PLOT(X,Y1,3)
00300          .  .  .  .  CALL PLOT(X,Y1+.133,2)
00301          .  .  .  ...FIN
00302          .  .  .  (2)
00303          .  .  .  .  JDD = 1
00304          .  .  .  .  CALL PLOT(X,Y1,3)
00305          .  .  .  .  CALL PLOT(X,Y2,2)
00306          .  .  .  ...FIN
00307          .  .  ...FIN
00308          .  ...FIN
00309          ...FIN
00310    C---DRAW THE GRAPH
00311          SIZ2  =  0.07
00312          NABSV = 0
00313          CALL NEWPEN(1)
00314          I1    = 0
00315          I2    = 0
00316          NDEL  = 0
00317          DO (JK=1,NTOT)
00318          .  Y    = H(JK)
00319          .  IYR  = T(JK)/10000.
00320          .  MO   = T(JK)/100.-FLOAT(IYR)*100.
00321          .  IDA  = T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
00322          .  DAYS = (FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
00323          .  X    = (DAYS*XSCALE+FLOAT(IMN))/100.
00324          .  WHEN (Y .LE. 0.)
00325          .  .  Y    = FLOAT(JMN)/100.
00326          .  .  NDEL = NDEL+1
00327          .  .  IDEL(NDEL) = JK
00328          .  ...FIN
00329          .  ELSE
00330          .  .  Y = (JMN+(ALOG10(Y)-MINY)*IYDF)/100.
00331          .  ...FIN
00332    C     .  CHECK FOR POINT WITHIN REGION
00333          .  IF (X .LT. X1 .OR. X. GT. X2) GO TO 40
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:11  PAGE  00007

00334          .  IF (Y .LT. Y1 .OR. Y. GT. Y2) GO TO 40
00335          .  IF (NDEL .NE. 0)
00336    C  SEE IF POINT HAS BEEN DELETED
00337          .  .  DO (J=1,NDEL)
00338          .  .  .  IF (IDEL(J) .EQ. JK) GO TO 3820
00339          .  .  ...FIN
00340          .  ...FIN
00341    C  NO
00342          .  I1    = I1+1
00343          .  T(I1) = X
00344          .  H(I1) = Y
00345          .  GO TO 40
00346    C  YES---STORE IN SECONDARY FILE
00347    3820  .  I2 = I2+1
00348          .  XDEL(I2) = X
00349          .  YDEL(I2) = Y
00350    40    .  CONTINUE
00351          ...FIN
00352    C
00353    C  PLOT DATA POINTS
00354          SIZ = 0.1
00355          CALL PLOT(T(1),H(1),3)
00356          LP  = -1
00357          IF (JPNT .EQ. 1) LP = -2
00358          DO (I=1,I1)
00359          .  CALL SYMBOL(T(I),H(I),SIZ,1,0.,LP)
00360          ...FIN
00361          IF (I2 .NE. 0)
00362    C  PLOT DELETED POINTS
00363          .  DO (I=1,I2)
00364          .  .  CALL SYMBOL(XDEL(I),YDEL(I),SIZ,1,0.,-1)
00365          .  ...FIN
00366          ...FIN
00367    C
00368          CALL FACTOR(1.0)
00369          CALL NEWPEN(1)
00370          CALL MAPPLT (XXC,YYC,NLDES,H,T,MAPTYP)
00371          CALL PLOT(0.0,11.0,-3)
00372    45    CALL PLOTND
00373    46    IF (ICNTRL. EQ .1)
00374          .  IEFN = 33
00375          .  CALL SETEF(IEFN,IDS)
00376          ...FIN
00377          CALL EXIT

       ----------------------------------------

00378          TO FIND-MINY-MAXY
00379          .  AL1  = ALOG10(HMIN)
00380          .  IF (AL1 .GE. 0.) AL1 = AL1+1.
00381          .  AL2  = ALOG10(HMAX)
00382          .  IF (AL2 .GE. 0.) AL2 = AL2+1.
00383          .  MINY = IFIX(AL1)-1
00384          .  MAXY = IFIX(AL2)
00385          ...FIN
00386          END
```

PROCEDURE CROSS-REFERENCE TABLE

00378 FIND-MINY-MAXY
00123 00140

(FLECS VERSION 22.46)

CTMLPR.FLX

CONTAMINATION (LINE PRINTER)

```
00001    C ***************  DR0:[351,100]CTMLPR.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: SRV-11-WEL-1-2  ABREVIATED WELL DESIG, VERSION
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C     THIS PROGRAM IS DESIGNED TO RETRIEVE WELL CONTAMINANT DATA FROM THE
00008    C     CIRMIS DATA BANK AND PRINT THE DATA IN FORMATED FORM ON THE LINE
00009    C     PRINTER.  THIS PROGRAM IS STARTED BY 'MNTR11' AND RECEIVES CONTROL
00010    C     DATA VIA THE MEMORY TO MEMORY.
00011    C
00012    C  DATA FILES:
00013    C     NAME   LUN  TYPE  ACCESS
00014    C     FILE Q---FRMTCTM 1  RAN  R
00015    C
00016    C
00017    C  LOADING SEQUENCE:
00018    C
00019    C     MCR>TKB @DB0:[351,100]CTMLPR
00020    C
00021    C
00022    C  BATTELLE MEMORIAL INSTITUTE
00023    C  PACIFIC NORTHWEST LABORATORIES
00024    C  WATER & LAND RESOURCES DEPT.
00025    C
00026    C  AUTHOR(S):   SW AHLSTROM
00027    C     DR FRIEDRICHS
00028    C     SE WISE
00029    C
00030    C  DATE:        INITIAL VERSION  AUGUST  1974
00031    C     CURRENT VERSION  OCTOBER 1978
00032    C
00033    C
00034    C
00035          BYTE STAT,WLDES
00036          BYTE DT(500,11),AMON(3,12),YR(2),DAY(2),BLANK,LT(512),ASTRX
00037    C
00038          INTEGER*4 CTTADR(36), CTXADR(36)
00039          DIMENSION H(512),T(512),INODE(255),HINT(512)
00040          DIMENSION IBUF(253), KBUF(256)
00041    C
00042          COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
00043         1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
00044         2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
00045    C
00046          EQUIVALENCE (WLDES(1),IBUF(1))
00047          EQUIVALENCE (IDM(1),INODE(1)), (CTTADR(1),KBUF(1))
00048          EQUIVALENCE (CTXADR(1),KBUF(73))
00049    C
00050          DATA AMON/'J','A','N','F','E','B','M','A','R','A','P','R',
00051         1 'M','A','Y','J','U','N','J','U','L','A','U','G','S','E','P',
00052         2 'O','C','T','N','O','V','D','E','C'/
00053    C
```

```
00054          DATA BLANK/' '/
00055          DATA ASTRX/'*'/
00056    C
00057    C
00058    C *** READ THE TRANSFER NODE ***
00059    C      TASK = RAD50('MNTR45')
00060          CALL VRECSP (TASK,INODE,255,,,IOS)
00061          WRITE (5,399) WLDES,ICTM
00062    399   FORMAT (' ENTER CTMLPR',12A1,' CTM = ',I5)
00063    C      IBUF(200) IS A FLAG, 5=TI: TERMINAL, ANYTHING ELSE
00064    C      WILL SEND OUTPUT TO THE LINEPRINTER, DESIGNED FOR
00065    C      ACCESS BY ARHCO TERMINAL
00066          LUN1 = 6
00067          IF (IBUF(200).EQ.5) LUN1=5
00068    C
00069          CALL ASNLUN(1,IDISC,IUNIT)
00070          CALL OPFILE(1,'FRMTCTM',37000,,IPNT1)
00071          CALL OPFILE(1,'CTMHDR',6000.,ICHDR)
00072    C
00073    C *** RETRIEVE THE CONTAMINATION DATA ***
00074    C
00075          QADR = IBUF(61)
00076          CALL DPR(ICHDR,QADR,KBUF,256.)
00077    C
00078          IOFF = (ICTM-1)*3
00079          NPTS = KBUF(145+IOFF)
00080          DKSAV= CTIADR(ICTM)
00081          DKSEX= CTXADR(ICTM)
00082          NPSEX= KBUF(146+IOFF)
00083          INALOC=KBUF(147+IOFF)
00084    C
00085    C
00086          WRDI = 2*NPTS
00087          WI   = INALOC*256
00088          DKT  = DKSAV+INALOC
00089          DKM  = DKT+INALOC
00090          CALL DPR(IPNT1,DKSAV,H,WRDI)
00091          CALL DPR(IPNT1,DKT,T,WRDI)
00092          CALL DPR(IPNT1,DKM,MINT,WI)
00093          IF (NPSEX .NE. 0)
00094    C     .
00095    C    READ EXTENDED AREA (IF ANY)
00096    C     .
00097          .  WRDE = 2*NPSEX
00098          .  DKTE = DKSEX+2.
00099          .  DKME = DKTE+2.
00100          .  NPX  = NPTS+1
00101          .  CALL DPR(IPNT1,DKSEX,H(NPX),WRDE)
00102          .  CALL DPR(IPNT1,DKTE,T(NPX),WRDE)
00103          .  CALL DPR(IPNT1,DKME,MINT(NPX),256.)
00104          ...FIN
00105          NTOT  = NPTS+NPSEX
00106    C    REMOVE ANY MINUS VALUES AND SET THE PROPER .LT. INDICATOR
00107          DO (I=1,NTOT)
00108          .  LT(I)=BLANK
00109          .  IF(H(I) .LT. 0) LT(I)=ASTRX
```

```
00110           .  H(I)=ABS(H(I))
00111           ...FIN
00112     C
00113     C *** WRITE HEADER INFORMATION TO THE PRINTER ***
00114     C
00115           WRITE(LUN1,500) WLDES
00116     500   FORMAT(1H1,16X,'CONTAMINANT DATA - WELL NO. ',12A1//)
00117           SELECT (ICTM)
00118           .  (1)
00119     21    .  .  WRITE(LUN1,521)
00120     521   .  .  FORMAT(21X,'CONTAMINANT TYPE --  TOTAL ALPHA')
00121           .  ...FIN
00122           .  (2)
00123           .  .  WRITE(LUN1,522)
00124     522   .  .  FORMAT(21X,'CONTAMINANT TYPE --  TOTAL BETA')
00125           .  ...FIN
00126           .  (3)
00127           .  .  WRITE(LUN1,523)
00128     523   .  .  FORMAT(21X,'CONTAMINANT TYPE --  TRITIUM')
00129           .  ...FIN
00130           .  (4)
00131     24    .  .  WRITE(LUN1,524)
00132     524   .  .  FORMAT(21X,'CONTAMINANT TYPE --  NITRATE')
00133           .  ...FIN
00134           .  (5)
00135           .  .  WRITE(LUN1,525)
00136     525   .  .  FORMAT(21X,'CONTAMINANT TYPE --  STRONTIUM-90')
00137           .  ...FIN
00138           .  (6)
00139           .  .  WRITE(LUN1,526)
00140     526   .  .  FORMAT(21X,'CONTAMINANT TYPE --  CESIUM-137')
00141           .  ...FIN
00142           .  (7)
00143           .  .  WRITE(LUN1,527)
00144     527   .  .  FORMAT(21X,'CONTAMINANT TYPE --  COBALT-60')
00145           .  ...FIN
00146           .  (8)
00147     28    .  .  WRITE(LUN1,528)
00148     528   .  .  FORMAT(21X,'CONTAMINANT TYPE --  URANIUM-238')
00149           .  ...FIN
00150           .  (9)
00151           .  .  WRITE(LUN1,529)
00152     529   .  .  FORMAT(21X,'CONTAMINANT TYPE --  PLUTONIUM-239')
00153           .  ...FIN
00154           .  (10)
00155           .  .  WRITE(LUN1,530)
00156     530   .  .  FORMAT(21X,'CONTAMINANT TYPE --  RUTHENIUM-106')
00157           .  ...FIN
00158           .  (11)
00159           .  .  WRITE(LUN1,531)
00160     531   .  .  FORMAT(21X,'CONTAMINANT TYPE --  CHROMIUM')
00161           .  ...FIN
00162           .  (12)
00163           .  .  WRITE(LUN1,532)
00164     532   .  .  FORMAT(21X,'CONTAMINANT TYPE --  FLUORIDE')
00165           .  ...FIN
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:12:37  PAGE  00004

00166          .  (13)
00167          .  .  WRITE(LUN1,533)
00168   533    .  .  FORMAT(21X,'CONTAMINANT TYPE --  MAGNESIUM')
00169          .  ...FIN
00170          .  (14)
00171          .  .  WRITE(LUN1,534)
00172   534    .  .  FORMAT(21X,'CONTAMINANT TYPE --  IRON')
00173          .  ...FIN
00174          .  (15)
00175          .  .  WRITE(LUN1,535)
00176   535    .  .  FORMAT(21X,'CONTAMINANT TYPE --  PHOSPHATE')
00177          .  ...FIN
00178          .  (16)
00179          .  .  WRITE(LUN1,536)
00180   536    .  .  FORMAT(21X,'CONTAMINANT TYPE --  CHLORIDE')
00181          .  ...FIN
00182          .  (17)
00183          .  .  WRITE(LUN1,537)
00184   537    .  .  FORMAT(21X,'CONTAMINANT TYPE --  COPPER')
00185          .  ...FIN
00186          .  (18)
00187          .  .  WRITE(LUN1,538)
00188   538    .  .  FORMAT(21X,'CONTAMINANT TYPE --  HARDNESS')
00189          .  ...FIN
00190          .  (19)
00191          .  .  WRITE(LUN1,539)
00192   539    .  .  FORMAT(21X,'CONTAMINANT TYPE --  SOLIDS')
00193          .  ...FIN
00194          .  (20)
00195          .  .  WRITE(LUN1,540)
00196   540    .  .  FORMAT(21X,'CONTAMINANT TYPE --  MANGANESE')
00197          .  ...FIN
00198          .  (21)
00199          .  .  WRITE(LUN1,541)
00200   541    .  .  FORMAT(21X,'CONTAMINANT TYPE --  TOTAL ORGANIC CARBON')
00201          .  ...FIN
00202          .  (22)
00203          .  .  WRITE(LUN1,542)
00204   542    .  .  FORMAT(21X,'CONTAMINANT TYPE --  PH')
00205          .  ...FIN
00206          .  (23)
00207          .  .  WRITE(LUN1,543)
00208   543    .  .  FORMAT(21X,'CONTAMINANT TYPE --  SULFATE')
00209          .  ...FIN
00210          .  (24)
00211          .  .  WRITE(LUN1,544)
00212   544    .  .  FORMAT(21X,'CONTAMINANT TYPE --  SODIUM')
00213          .  ...FIN
00214          .  (25)
00215          .  .  WRITE(LUN1,545)
00216   545    .  .  FORMAT(21X,'CONTAMINANT TYPE --  CALCIUM')
00217          .  ...FIN
00218          .  (26)
00219          .  .  WRITE(LUN1,546)
00220   546    .  .  FORMAT(21X,'CONTAMINANT TYPE --  TOTAL GAMMA')
00221          .  ...FIN
```

```
00222          .  (27)
00223          .  .  WRITE(LUN1,547)
00224    547   .  .  FORMAT(21X,'CONTAMINANT TYPE --  BICARBONATE ION')
00225          .  ...FIN
00226          .  (28)
00227          .  .  WRITE(LUN1,548)
00228    548   .  .  FORMAT(21X,'CONTAMINANT TYPE --  CARBONATE ION')
00229          .  ...FIN
00230          .  (29)
00231          .  .  WRITE(LUN1,549)
00232    549   .  .  FORMAT(21X,'CONTAMINANT TYPE --  TOTAL POTASSIUM')
00233          .  ...FIN
00234          .  (30)
00235          .  .  WRITE(LUN1,551)
00236    551   .  .  FORMAT(21X,'CONTAMINANT TYPE --  SPECIFIC CONDUCTIVITY')
00237          .  ...FIN
00238          .  (31)
00239          .  .  WRITE(LUN1,552)
00240    552   .  .  FORMAT(21X,'CONTAMINANT TYPE --  BORON')
00241          .  ...FIN
00242          .  (32)
00243          .  .  WRITE(LUN1,553)
00244    553   .  .  FORMAT(21X,'CONTAMINANT TYPE --  LOW ALPHA')
00245          .  ...FIN
00246          ...FIN
00247    C
00248          WHEN (NEX. EQ . 99)
00249          .  J = 0
00250          .  DO (I = 1,NTOT)
00251          .  .  NYR = T(I)/10000.
00252          .  .  IF (NYR.GE.MINX. AND .NYR.LE.MAXX)
00253          .  .  .  J = J+1
00254          .  .  .  T(J) = T(I)
00255          .  .  .  H(J) = H(I)
00256          .  .  .  MINT(J) = MINT(I)
00257          .  .  ...FIN
00258          .  ...FIN
00259          .  NTOT = J
00260          .  WRITE (LUN1,560) ICTM,MINX,MAXX,NTOT
00261    560   .  FORMAT(21X,'CONTAMINANT CODE =',I2/
00262         1.   21X,'MEASURMENTS (CY 19',I2,'-19',I2,') =',I3//)
00263          ..FIN
00264          ELSE
00265          .  WRITE(LUN1,561) ICTM,NTOT
00266    561   .  FORMAT(21X,'CONTAMINANT CODE =',I2/
00267         2.   21X,'MEASURMENTS TO DATE = ',I3//)
00268          ..FIN
00269          IF (NTOT.LE.0) GO TO 99
00270    C
00271    C *** CONVERT THE DATE TO ASCII ***
00272    C
00273          DO (I=1,NTOT)
00274          .  DT(I,3)='-'
00275          .  DT(I,7)='-'
00276          .  IYR=T(I)/10000.
00277          .  IMON=T(I)/100.-IYR*100.
```

```
(FLECS VERSION 22.46)     11-SEP-79   14112137  PAGE  00006

00278           .  WHEN (IMON.GT.0)
00279           .  .  IDAY=T(I)-IMON*100.-IYR*10000.
00280           .  .  DT(I,4)=AMON(1,IMON)
00281           .  .  DT(I,5)=AMON(2,IMON)
00282           .  .  DT(I,6)=AMON(3,IMON)
00283           .  .  ENCODE (2,501,YR) IYR
00284    501    .  .  FORMAT(I2)
00285           .  .  ENCODE (2,501,DAY) IDAY
00286           .  .  DT(I,1)=DAY(1)
00287           .  .  DT(I,2)=DAY(2)
00288           .  .  DT(I,8)=YR(1)
00289           .  .  DT(I,9)=YR(2)
00290           .  .  DT(I,10)=MINT(I)/60
00291           .  .  DT(I,11)=MOD(MINT(I),60)
00292           .  ...FIN
00293           .  ELSE
00294           .  .  DO (M=1,9) DT(I,M)=' '
00295           .  .  DT(I,10)=0
00296           .  .  DT(I,11)=0
00297           .  ...FIN
00298           ...FIN
00299    C
00300    C *** DUMP THE DATA TO THE PRINTER ***
00301    C
00302           IF (NTOT .GT. 0)
00303           .  WRITE(LUN1,502) ((DT(K,M),M=1,11),H(K),LT(K),K=1,NTOT)
00304    502    .  FORMAT(3(1X,9A1,1X,2I2,E9.2,A1,2X))
00305           ...FIN
00306    99     IF (ICNTRL. EQ .1)
00307           .  IEFN = 33
00308           .  CALL SETEF (IEFN,IDS)
00309           .  WRITE (5,599) IEFN,IDS
00310    599    .  FORMAT (' IEFN,IDS =',2I5)
00311           ...FIN
00312           CALL EXIT
00313           END

          (FLECS VERSION 22.46)

          ----------------------------------------
```

CTMGLD.FLX

CONTAMINATION (GOULD)

```
C ***************  [351,100]CTMGLD.FLX  ***************
C
C
C  PROGRAM ID: SRV-11-WEL-1-3
C
C  PROGRAM DESCRIPTION:
C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL CONTAMINATE DATA FROM THE
C     CIRMIS DATA BANK AND DRAW A GOULD PLOT OF THE DATA.  THIS PROGRAM
C     IS STARTED BY 'MNTR11' AND RECEIVES CONTROL DATA VIA THE TRANSFER
C     NODE.
C
C  DATA FILES:
C        NAME                        LUN              TYPE             ACCESS
C        FILE Q---FRMTCTM             1               RAN               R
C
C
C  LOADING SEQUENCE:
C
C     MCR>TKB @[351,100]CTMGLD
C
C
C  BATTELLE MEMORIAL INSTITUTE
C  PACIFIC NORTHWEST LABORATORIES
C  WATER & LAND RESOURCES DEPT.
C
C  AUTHOR(S):    DR FRIEDRICHS
C
C  DATE:         INITIAL VERSION      JANUARY   1977
C                CURRENT VERSION      JANUARY   1979
C
C
      BYTE A(3),B(2),WLDES
C
      INTEGER BUFFER,SIZ,SIZ1,SIZ2,YC
      INTEGER*4 CTIADR(36), CTXADR(36)
C
      DIMENSION H(512),T(512),IBUF(253),XDEL(17),YDEL(17)
      DIMENSION INODE(255), KBUF(256)
C
      COMMON/BFR/ BUFFER(4800)
C
      COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
     1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
     2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
C
      EQUIVALENCE (WLDES(1),IBUF(1))
      EQUIVALENCE (IDM(1),INODE(1)), (CTIADR(1),KBUF(1))
      EQUIVALENCE (CTXADR(1),KBUF(73))
C
C *** READ THE TRANSFER NODE ***
C
C     TASK = RAD50('MNTR45')
      CALL VRECSP (TASK,INODE,255,,,IDS)
```

```
00054          JPNT = 0
00055          NDEL = 0
00056    C
00057          CALL ASNLUN(1,IDISC,IUNIT)
00058          CALL DPFILE(1,'FRMTCTM',37000,,IPNT1)
00059          CALL DPFILE(1,'CTMHDR',6000,,ICHDR)
00060    C
00061    C *** RETRIEVE THE CONTAMINATION DATA ***
00062    C
00063          DADR = IBUF(61)
00064          CALL DPR(ICHDR,DADR,KBUF,256,)
00065    C
00066          IOFF = (ICTM-1)*3
00067          NPTS = KBUF(145+IOFF)
00068          DKSAV= CTIADR(ICTM)
00069          DKSEX= CTXADR(ICTM)
00070          NPSEX= KBUF(146+IOFF)
00071          INALOC=KBUF(147+IOFF)
00072    C
00073    C
00074          WRDI = 2*NPTS
00075          WI   = INALOC*256
00076          DKT  = DKSAV+INALOC
00077          DKM  = DKT+INALOC
00078          CALL DPR(IPNT1,DKSAV,H,WRDI)
00079          CALL DPR(IPNT1,DKT,T,WRDI)
00080          IF (NPSEX .EQ. 0) GO TO 10
00081    C
00082    C     READ EXTENDED AREA (IF ANY)
00083    C
00084          WRDE = 2*NPSEX
00085          DKTE = DKSEX+2.
00086          DKME = DKTE+2.
00087          NPX  = NPTS+1
00088          CALL DPR(IPNT1,DKSEX,H(NPX),WRDE)
00089          CALL DPR(IPNT1,DKTE,T(NPX),WRDE)
00090    C    REMOVE ANY MINUS VALUES
00091    10    NTOT=NPTS+NPSEX
00092          DO 11 I=1,NTOT
00093          H(I)=ABS(H(I))
00094    11    CONTINUE
00095          IF (MINX .EQ. 0) NEX = 0
00096          CONDITIONAL
00097          .  (NEX .EQ. 0)
00098    C     .  .
00099    C *** CALCULATE AUTO SCALE FACTORS ***
00100    C     .  .
00101          .  .  TMAX = 0.0
00102          .  .  TMIN = 999999.
00103          .  .  HMAX = 0.0
00104          .  .  HMIN = 1.0E+20
00105          .  .  DO (I=1,NTOT)
00106          .  .  .  IF (T(I) .GT. TMAX) TMAX = T(I)
00107          .  .  .  IF (T(I) .LT. TMIN) TMIN = T(I)
00108          .  .  .  IF (H(I) .GT. 0.)
00109          .  .  .  .  IF (H(I) .GT. HMAX) HMAX = H(I)
```

```
00110          .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00111          .  .  .  ...FIN
00112          .  .  ...FIN
00113          .  .  MINX = IFIX(TMIN/10000.)
00114          .  .  MAXX = IFIX(TMAX/10000.)
00115          .  .  FIND=MINY=MAXY
00116          .  ...FIN
00117          .  (NEX .NE. 99)
00118          .  .  MINX = IBUF(232)
00119          .  .  MINY = IBUF(233)
00120          .  .  MAXX = IBUF(234)
00121          .  .  MAXY = IBUF(235)
00122          .  ...FIN
00123          .  (NEX .EQ. 99)
00124          .  .  MINX = IBUF(232)
00125          .  .  MAXX = IBUF(234)
00126          .  .  HMAX = 0.0
00127          .  .  HMIN = 1.0E+20
00128          .  .  J = 0
00129          .  .  DO (I=1,NTOT)
00130          .  .  .  NYR = T(I)/10000.
00131          .  .  .  IF (NYR .GE. IBUF(232) .AND. NYR .LE. IBUF(234))
00132          .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00133          .  .  .  .  IF( H(I) .GT. HMAX) HMAX = H(I)
00134          .  .  .  .  J = J + 1
00135          .  .  .  .  H(J) = H(I)
00136          .  .  .  .  T(J) = T(I)
00137          .  .  .  ...FIN
00138          .  .  ...FIN
00139          .  .  NTOT = J
00140          .  .  IF (J .LE. 0) GO TO 46
00141          .  .  IF (HMAX .EQ. 0.) GO TO 46
00142          .  .  FIND=MINY=MAXY
00143          .  ...FIN
00144          ...FIN
00145    14    IMN=170
00146          IMX=770
00147          JMN=300
00148          JMX=900
00149          SCLX=IMX-IMN
00150          SCLY=JMX-JMN
00151          SIZ=24
00152          SIZ1=16
00153          SIZ2=8
00154    C---DEFINE BACKGROUND AND PLOT IT
00155          X1=FLOAT(IMN)
00156          X2=FLOAT(IMX)
00157          Y1=FLOAT(JMN)
00158          Y2=FLOAT(JMX)
00159    C
00160    C *** DRAW THE PLOT ***
00161    C
00162          CALL DDTINI(7,BUFFER,32,1200)
00163          DO 9999 INDEX=1,864,32
00164    C
00165          CALL FEDCL(0.0,0.0,0.0,1)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:23  PAGE  00004

 00166   C---DEFINE BACKGROUND AND PLOT IT
 00167         CALL DOTLIN(IMN,JMN,IMN,JMX)
 00168         CALL DOTLIN(IMN,JMX,IMX,JMX)
 00169         CALL DOTLIN(IMX,JMX,IMX,JMN)
 00170         CALL DOTLIN(IMX,JMN,IMN,JMN)
 00171         IXS=120
 00172         IYS=425
 00173         CALL DOTCHR(IXS,IYS,'CONCENTRATION',,SIZ,1)
 00174         IYS=IYS+240
 00175         GO TO(15,15,15,16,15,15,15,17,15,15,16,16,16,
 00176        1      16,16,16,16,16,16,16,16,1750,16,16,16,15),ICTM
 00177   15    CALL DOTCHR(IXS,IYS,'(PCI/ML)',,SIZ,1)
 00178         GO TO 1750
 00179   16    CALL DOTCHR(IXS,IYS,'(MG/L)',,SIZ,1)
 00180         GO TO 1750
 00181   17    CALL DOTCHR(IXS,IYS,'(NG/ML)',,SIZ,1)
 00182   1750  CALL DOTCHR(360,250,'CALENDAR YEAR',,SIZ)
 00183         IX=240
 00184         IY=150
 00185         IZ=SIZ1
 00186         CALL DOTCHR(IX,IY,'CONCENTRATION HISTORY ---',,IZ)
 00187         IX=575
 00188         IF(ICTM.EQ.1) CALL DOTCHR(IX,IY,'TOTAL ALPHA',,IZ)
 00189         IF(ICTM.EQ.2) CALL DOTCHR(IX,IY,'TOTAL BETA',,IZ)
 00190         IF(ICTM.EQ.3) CALL DOTCHR(IX,IY,'TRITIUM',,IZ)
 00191         IF(ICTM.EQ.4) CALL DOTCHR(IX,IY,'NITRATE',,IZ)
 00192         IF(ICTM.EQ.5) CALL DOTCHR(IX,IY,'STRONTIUM-90',,IZ)
 00193         IF(ICTM.EQ.6) CALL DOTCHR(IX,IY,'CESIUM-137',,IZ)
 00194         IF(ICTM.EQ.7) CALL DOTCHR(IX,IY,'COBALT-60',,IZ)
 00195         IF(ICTM.EQ.8) CALL DOTCHR(IX,IY,'URANIUM-238',,IZ)
 00196         IF(ICTM.EQ.9) CALL DOTCHR(IX,IY,'PLUTONIUM-239',,IZ)
 00197         IF(ICTM.EQ.10)CALL DOTCHR(IX,IY,'RUTHENIUM-106',,IZ)
 00198         IF(ICTM.EQ.11)CALL DOTCHR(IX,IY,'CHROMIUM',,IZ)
 00199         IF(ICTM.EQ.12)CALL DOTCHR(IX,IY,'FLUORIDE',,IZ)
 00200         IF(ICTM.EQ.13)CALL DOTCHR(IX,IY,'MAGNESIUM',,IZ)
 00201         IF(ICTM.EQ.14)CALL DOTCHR(IX,IY,'IRON',,IZ)
 00202         IF(ICTM.EQ.15)CALL DOTCHR(IX,IY,'PHOSPHATE',,IZ)
 00203         IF(ICTM.EQ.16)CALL DOTCHR(IX,IY,'CHLORIDE',,IZ)
 00204         IF(ICTM.EQ.17)CALL DOTCHR(IX,IY,'COPPER',,IZ)
 00205         IF(ICTM.EQ.18)CALL DOTCHR(IX,IY,'HARDNESS',,IZ)
 00206         IF(ICTM.EQ.19)CALL DOTCHR(IX,IY,'SOLIDS',,IZ)
 00207         IF(ICTM.EQ.20)CALL DOTCHR(IX,IY,'MANGANESE',,IZ)
 00208         IF(ICTM.EQ.21)CALL DOTCHR(IX,IY,'TOTAL ORGANIC CARBON',,IZ)
 00209         IF(ICTM.EQ.22)CALL DOTCHR(IX,IY,'PH',,IZ)
 00210         IF(ICTM.EQ.23)CALL DOTCHR(IX,IY,'SULFATE',,IZ)
 00211         IF(ICTM.EQ.24)CALL DOTCHR(IX,IY,'SODIUM',,IZ)
 00212         IF(ICTM.EQ.25)CALL DOTCHR(IX,IY,'CALCIUM',,IZ)
 00213         IF(ICTM.EQ.26)CALL DOTCHR(IX,IY,'TOTAL GAMMA',,IZ)
 00214         IF(ICTM.EQ.27)CALL DOTCHR(IX,IY,'BICARBONATE ION',,IZ)
 00215         IF(ICTM.EQ.28)CALL DOTCHR(IX,IY,'CARBONATE ION',,IZ)
 00216         IF(ICTM.EQ.29)CALL DOTCHR(IX,IY,'TOTAL POTASSIUM',,IZ)
 00217         IF(ICTM.EQ.30)CALL DOTCHR(IX,IY,'SPECIFIC CONDUCTIVITY',,IZ)
 00218         IF(ICTM.EQ.31)CALL DOTCHR(IX,IY,'BORON',,IZ)
 00219         IF(ICTM.EQ.32)CALL DOTCHR(IX,IY,'LOW ALPHA',,IZ)
 00220         IXL=(SCLX/2.) - FLOAT(SIZ2)*.75*16. + X1
 00221         IY=130
```

```
00222          CALL DOTCHR(IXL,IY,'WELL DESIGNATION - ',,SIZ2)
00223          CALL DOTCHR(IXL+120,IY,WLDES,12,SIZ2)
00224          NDIF=MAXY-MINY
00225          YSCALE=SCLY/FLOAT(NDIF)
00226    C
00227    C
00228    C----DRAW Y LINES FOR CHART
00229          IYDF=(JMX-JMN)/NDIF
00230          DO 22 JD=1,NDIF
00231          YC=(JMN+(JD-1)*IYDF)
00232          NN=MINY+JD-1
00233          IF(JD.EQ.1) GO TO 21
00234          CALL DOTLIN(IMN,YC,IMX,YC)
00235    21    CALL DOTCHR(IMN-30,YC,'1.E',,SIZ2)
00236          ENCODE(3,621,A) NN
00237    621   FORMAT(I3)
00238          CALL DOTCHR(IMN-20,YC,A,3,ISIZ2)
00239          NB=1
00240          IF(JD.EQ.1) NB=2
00241          DO 22 K=NB,9
00242          IY=FLOAT(YC)+(ALOG10(FLOAT(K))*IYDF)
00243          CALL DOTLIN(IMN,IY,IMX,IY)
00244    22    CONTINUE
00245          CALL DOTCHR(IMN-30,JMX,'1.E',,SIZ2)
00246          ENCODE(3,621,A) MAXY
00247          CALL DOTCHR(IMN-20,JMX,A,3,SIZ2)
00248    C
00249    C
00250    C----DRAW X LINES FOR CHART
00251          NDIF=(MAXX-MINX) + 1
00252          XSCALE=SCLX/(FLOAT(NDIF)*365.)
00253          XINC=(SCLX/FLOAT(NDIF))
00254          XADJ=XINC/2. + FLOAT(SIZ2)*.75
00255          JDD=1
00256          DO 3650 JD=MINX,MAXX
00257          X=FLOAT(JD-MINX+1)*XINC + X1
00258          IX=X-XADJ
00259          ENCODE(2,636,B) JD
00260    636   FORMAT(I2)
00261          CALL DOTCHR(IX,JMN-20,B,2,SIZ2)
00262          IX=X
00263          IF(JD.EQ.MAXX) GO TO 3650
00264          GO TO(3620,3630),JDD
00265    3620  JDD=2
00266          CALL DOTLIN(IX,JMN,IX,JMX)
00267          GO TO 3650
00268    3630  JDD=1
00269          CALL DOTLIN(IX,JMN,IX,JMX)
00270    3650  CONTINUE
00271    C
00272    C
00273    C---DRAW THE GRAPH
00274    C
00275    C     IF(INDEX.EQ.1) WRITE(6,637) NTOT,(T(I),H(I),I=1,NTOT)
00276    637   FORMAT(' NTOT= ',I3,/,(1X,2F10.2))
00277          NARSV=0
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:23  PAGE  00006

 00278          I1=0
 00279          I2=0
 00280          DO 40 JK=1,NTOT
 00281    3701  Y=H(JK)
 00282          IYR=T(JK)/10000.
 00283          MO=T(JK)/100.-FLOAT(IYR)*100.
 00284          IDA=T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
 00285    38    DAYS=(FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
 00286          X=(DAYS*XSCALE+FLOAT(IMN))
 00287          IF(Y.GT.0.) GO TO 3805
 00288          Y=FLOAT(JMN)
 00289          NDEL=NDEL+1
 00290          IDEL(NDEL)=JK
 00291          GO TO 3806
 00292    3805  Y=(JMN+(ALOG10(Y)-MINY)*IYOF)
 00293    C     CHECK FOR POINT WITHIN REGION
 00294    3806  IF(X.LT.X1.OR.X.GT.X2) GO TO 40
 00295          IF(Y.LT.Y1.OR.Y.GT.Y2) GO TO 40
 00296          IF(NDEL.EQ.0) GO TO 3810
 00297    C  SEE IF POINT HAS BEEN DELETED
 00298          DO 3810 J=1,NDEL
 00299          IF(IDEL(J).EQ.JK) GO TO 3820
 00300    3810  CONTINUE
 00301    C  NO---PLOT DATA POINT
 00302          IX=X
 00303          IY=Y
 00304          CALL DOTCHR(IX,IY,'*',,SIZ2)
 00305          GO TO 40
 00306    C  YES---PLOT DELETED POINT
 00307    3820  IX=X
 00308          IY=Y
 00309          CALL DOTCHR(IX,IY,'O',,SIZ2)
 00310    40    CONTINUE
 00311    C
 00312    C  PLOT DATA POINTS
 00313    9990  CALL DOTOUT
 00314          CALL DOTDUN
 00315    46    IF (ICNTRL.NE.1) CALL EXIT
 00316          IFFN = 33
 00317          CALL SETEF(IEFN,IOS)
 00318          CALL EXIT

        ----------------------------------------

 00319          TO FIND=MINY=MAXY
 00320          .  AL1  = ALOG10(HMIN)
 00321          .  IF (AL1 .GE. 0.) AL1 = AL1+1.
 00322          .  AL2  = ALOG10(HMAX)
 00323          .  IF (AL2 .GE. 0.) AL2 = AL2+1.
 00324          .  MINY = IFIX(AL1)-1
 00325          .  MAXY = IFIX(AL2)
 00326          ...FIN
 00327          END

        ----------------------------------------
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:13:23  PAGE  00007

                PROCEDURE CROSS-REFERENCE TABLE

          00319 FIND-MINY-MAXY
                00115  00142


          (FLECS VERSION 22.46)

          ----------------------------------------

  00328   C
  00329   C
  00330           SUBROUTINE FEDCL(XX,YY,IR)
  00331   C
  00332           INTEGER X,Y,BUFFER
  00333   C
  00334           COMMON/BFR/ BUFFER(4800)
  00335   C
  00336   C
  00337           XMAX=850.
  00338           YMAX=1100.
  00339           IF(IR.EQ.1) GO TO 10
  00340           XMAX=1100.
  00341           YMAX=850.
  00342   10      Y=YY
  00343           DO 15 J=1,4
  00344           X=XX
  00345           IF(J.GE.3) Y=YY+YMAX
  00346           IF(J.EQ.2.OR.J.EQ.4) X=XX+XMAX
  00347           CALL DOTLIN(X,Y+10,X+20,Y+10)
  00348           CALL DOTLIN(X+10,Y,X+10,Y+20)
  00349   15      CONTINUE
  00350           RETURN
  00351           END

          (FLECS VERSION 22.46)

          ----------------------------------------
```

WSTCAL.FTN

WELL STRUCTURES (CALCOMP)

```
FORTRAN IV-PLUS V02-510          14:15:33    11-SEP-79          PAGE 1
WSTCAL,FTN       /TR:BLOCKS/WR

        C
        C ***************  [351,100]WSTCAL,FTN  ****************
        C
        C
        C  PROGRAM ID:   CIRMIS-11-ONL-2-1
        C
        C  PROGRAM DESCRIPTION:
        C                        THIS PROGRAM RETRIEVES PERFORATION DATA
        C                        FROM "FRMTWST" AND RELAYS IT
        C                        TO THE CALCOMP PLOTTER
        C
        C  DATA FILES:
        C       NAME                        LUN              TYPE             ACCESS
        C       FILE 0---FRMTWST            1                RAN              R
        C
        C
        C  LOADING SEQUENCE:
        C                        TKB>@[106,215]WSTCAL
        C
        C
        C  BATTELLE MEMORIAL INSTITUTE
        C  PACIFIC NORTHWEST LABORATORIES
        C  WATER & LAND RESOURCES DEPT.
        C
        C  AUTHOR(S):       DD HOSTETLER
        C
        C  DATE:          INITIAL VERSION   AUGUST 1974
        C                 CURRENT VERSION   MAY 1979
        C
        C
        C
0001            BYTE STAT, WDES, WLDES(2)
        C
0002            REAL ITEMP,NO,ICRMT,LCAR,MAX,MIN
0003            INTEGER BUFF,BUF1,BUF2,BUF3,BUF4,BUF5
        C
0004            DIMENSION INODE(255),BUFF(768),HEAD(3,2)
0005            DIMENSION DNAM(6),PERF(50),SCREEN(20),CASING(300)
0006            DIMENSION PIEZ(18),COMMNT(54),COMT(8),DIA(30)
0007            DIMENSION BUF1(18),BUF2(10),BUF3(8),BUF4(32),BUF5(8)
0008            DIMENSION ITRAN(253)
        C
0009            COMMON/T1/DIAM,CAS(8)
0010            COMMON/T2/PF(5)
0011            COMMON/T3/S(4)
0012            COMMON/T4/CM(16)
0013            COMMON/T5/PZ(4)
0014            COMMON/WOR/ IDM(2),WDES(12),IDM1(28),IDKNT,NSTART,
                1           NWORD,IDM2(213),IDISC,IUNIT,ICNTRL
        C
0015            EQUIVALENCE (BUF1(1),DIAM),(BUF2(1),PF(1))
0016            EQUIVALENCE (BUF3(1),S(1)),(BUF4(1),CM(1))
0017            EQUIVALENCE (BUF5(1),PZ(1))
0018            EQUIVALENCE (WDES(1),INODE(1)),(IDM(1),ITRAN(1))
        C
0019            DATA LCAR/1H</
```

```
0020            DATA RCAR/1H>/
0021            DATA BLANK/4H NG /
0022            DATA YES/3HYES/
0023            DATA NO/2HNO/
0024            DATA REG/4HREGU/
0025            DATA STAG/4HSTAG/
0026            DATA SPIR/4HSPIR/
0027            DATA SLANT/1HX/
0028            DATA BLANKS/2H0,/
0029            DATA PERFER/1HP/
0030            DATA B27/4H 11 /
0031            DATA B29/4H 12 /
0032            DATA WDC1/4H 12"/
0033            DATA WDC2/4H 10"/
0034            DATA WDC3/4H 24"/
0035            DATA WDC4/4H 8" /
0036            DATA WDC5/4H 36"/
0037            DATA WDC6/4H 9" /
0038            DATA WDC7/4H 5" /
0039            DATA WDC8/4H 18"/
0040            DATA WDC9/4H 16"/
0041            DATA WDC10/4H 6" /
0042            DATA HEAD(1,1)/4HHOLE/
0043            DATA HEAD(2,1)/4HSPAC/
0044            DATA HEAD(3,1)/4HTYPE/
0045            DATA B10/2H10/
0046            DATA B4/4H  4 /
      C
0047            N=1
0048            M=1
0049            INC=0
0050            INP=0
0051            INS=0
0052            INCM=0
0053            INPE=0
0054            SCRTOT=0.0
      C   ZERO ARRAYS
0055            DO 5 I=1,250
0056            IF(I.GT.50) GO TO 4
0057            IF(I.GT.20) GO TO 3
0058            IF(I.GT.16) GO TO 2
0059            PIEZ(I)=0.
0060      2     SCREEN(I)=0.
0061      3     PERF(I)=0.
0062      4     CASING(I)=0.
0063      5     CONTINUE
      C
      C***READ TRANSFER NODE***
      C
      C       TASK = RAD50('MNTR45')
0064            CALL VRECSP (TASK,ITRAN,255,,,IOS)
      C
0065            CALL ASNLUN(1,IDISC,IUNIT)
0066            CALL OPFILE(1,'FRMTWST',1000,,IPNT1)
      C
      C***RETRIEVE DATA FROM BASE***
```

```
      C
0067            STRT=IDKNT
0068            CALL DPR(IPNT1,STRT,BUFF,768,)
0069            I=0
0070            NTOP=NSTART+NWORD-1
0071            DO 1111 J=NSTART,NTOP
0072            I=I+1
0073            BUFF(I)=BUFF(J)
0074    1111    CONTINUE
      C
0075            KNT=0
0076    1010    KNT=KNT+1
0077            IF(KNT.GT.NWORD) GO TO 99
0078            I1=BUFF(KNT)
0079            GO TO (1000,2000,3000,4000,5000),I1
      C
      C----READ 1 CARDS
      C
0080    1000    DO 11 I=1,18
0081            KNT=KNT+1
0082            BUF1(I)=BUFF(KNT)
0083    11      CONTINUE
0084            GO TO 81
      C----READ 2 CARDS
0085    2000    DO 21 I=1,10
0086            KNT=KNT+1
0087            BUF2(I)=BUFF(KNT)
0088    21      CONTINUE
0089            GO TO 82
      C----READ 3 CARDS
0090    3000    DO 31 I=1,8
0091            KNT=KNT+1
0092            BUF3(I)=BUFF(KNT)
0093    31      CONTINUE
0094            GO TO 83
      C----READ 4 CARDS
0095    4000    DO 41 I=1,32
0096            KNT=KNT+1
0097            BUF4(I)=BUFF(KNT)
0098    41      CONTINUE
0099            GO TO 84
      C----READ 5 CARDS
0100    5000    DO 51 I=1,8
0101            KNT=KNT+1
0102            BUF5(I)=BUFF(KNT)
0103    51      CONTINUE
0104            GO TO 85
      C
      C***STORE DATA***
      C
0105    81      DO 8100 J=1,8
0106            IF(ABS(CAS(J)).LE.0.01) GO TO 8100
0107            INC=INC+1
0108            CASING(INC)=CAS(J)
0109    8100    CONTINUE
0110            DIA(M)=DIAM
```

```
0111              M=M+1
0112              GO TO 1010
0113     82       DO 8200 J=1,5
0114              INP=INP+1
0115     8200     PERF(INP)=PF(J)
0116              GO TO 1010
0117     83       DO 8300 J=1,4
0118              IF(J.NE.2) GO TO 8301
0119              IF(S(J).GT.SCRTOT) SCRTOT=S(J)
0120     8301     INS=INS+1
0121              SCREEN(INS)=S(J)
0122     8300     CONTINUE
0123              GO TO 1010
0124     84       DO 8400 J=1,16
0125              INCM=INCM+1
0126     8400     COMMNT(INCM)=CM(J)
0127              GO TO 1010
0128     85       DO 8500 J=2,4
0129              INPE=INPE+1
0130     8500     PIEZ(INPE)=PZ(J)
0131              DNAM(N)=PZ(1)
0132              N=N+1
0133              GO TO 1010
         C
         C
         C  ********* SETUP WELL SPECS ***********
         C
         C  **ELIMINATE UNUSED PART OF WELL**
         C
0134     99       MIN=2000.
0135              MAX=0.
0136              IF(INP.EQ.0) GO TO 6001
0137              DO 6 J=1,INP,5
0138              IF(PERF(J+1).GT.MAX) MAX=PERF(J)
0139              IF(PERF(J).LT.MIN) MIN=PERF(J)
0140     6        CONTINUE
0141     6001     IF(INS.EQ.0) GO TO 7001
0142              DO 7 J=1,INS,4
0143              IF(SCREEN(J+1).GT.MAX) MAX=SCREEN(J)
0144              IF(SCREEN(J).LT.MIN) MIN=SCREEN(J)
0145     7        CONTINUE
0146     7001     IF(INPE.EQ.0) GO TO 8001
0147              DO 8 J=1,INPE,3
0148              IF(PIEZ(J).GT.MAX) MAX=PIEZ(J)
0149              IF(PIEZ(J).LT.MIN) MIN=PIEZ(J)
0150     8        CONTINUE
0151     8001     IF(INC.EQ.0) GO TO 9001
0152              DO 9 J=2,INC,2
0153              IF(CASING(J).GT.MAX) MAX=CASING(J)
0154     9        CONTINUE
0155     9001     END=MAX
0156              BEGIN=MIN-20.
0157              ISIDE=1
0158              RANGE=END-BEGIN
0159              SCALE=14./RANGE
         C
```

```
        C
        C  ********** DRAW WELL **********
        C
        C
0160            CALL PLOTS(0.,0.,7)
0161            CALL NEWPEN(1)
0162            CALL FACTOR(1.)
0163            CALL PLOT(0.,4.0,-3)
0164            CALL SYMBOL(2.,15.4,.22,'WELL DESIGNATION',0.,16)
0165            CALL SYMBOL(2.8,15.1,.15,WDES,0.,12)
0166            IF(CASING(INC).GE.1.0) GO TO 9002
        C  CASING ERROR
0167            CALL SYMBOL(2.6,14.5,.1,'CASING ERROR,  CHECK DATA',0.,25)
0168            GO TO 98
0169    9002    YCASE=14.-(CASING(INC)-BEGIN)*SCALE
0170            YYY=YCASE
0171            IF(SCRTOT.GT.CASING(INC)) YYY=14.-(SCRTOT-BEGIN)*SCALE
0172            CALL PLOT(3.,YYY,3)
0173            CALL PLOT(3.,14.,2)
0174            CALL PLOT(3.,14.22,3)
0175            CALL PLOT(3.,14.55,2)
0176            CALL PLOT(4.0,14.55,3)
0177            CALL PLOT(4.0,14.22,2)
0178            CALL PLOT(4.0,14.,3)
0179            CALL PLOT(4.0,YYY,2)
0180            CALL PLOT(2.6,14.22,3)
0181            CALL PLOT(4.4,14.22,2)
0182            CALL PLOT(4.4,14.,3)
0183            CALL PLOT(2.6,14.,2)
        C  **GROUND LEVEL**
0184            CALL PLOT(0.,14.4,3)
0185            CALL PLOT(3.,14.4,2)
0186            CALL PLOT(4.0,14.4,3)
0187            CALL PLOT(7.0,14.4,2)
        C
        C
        C  **SETUP CASING DATA **
0188            SIZ1=.06
0189            PULLED=0.
0190            CTOTAL=0.
0191            IF(INC.EQ.0)GO TO 1001
0192            CALL PLOT(2.8,YCASE,3)
0193            CALL PLOT(4.5,YCASE,2)
0194            CALL NUMBER(4.7,YCASE,SIZ1,CASING(INC),0.,2)
0195            CALL SYMBOL(4.6,YCASE-.1,SIZ1,'(CASING END)',0.,12)
0196            DO 10 K=1,INC,2
0197            CTOTAL=CTOTAL+CASING(K)
0198            IF(CASING(K).GE.0.) GO TO 9010
        C  CASING PULLED BACK
0199            PULLED=PULLED+ABS(CASING(K))
0200    9010    IF(CTOTAL.LT.BEGIN) GO TO 10
0201            IF(CTOTAL.GT.END)GO TO 10
0202            IF(CASING(K).EQ.0.)GO TO 10
0203            Y=14.-(CTOTAL-BEGIN)*SCALE
0204            DEPTH=CTOTAL
0205            GO TO(951,952),ISIDE
```

```
0206     951      CALL PLOT(3.,Y,3)
0207              CALL PLOT(4.0,Y,2)
0208              CALL NUMBER(4.15,Y,.1Z1,DEPTH,0.,0)
0209              ISIDE=2
0210              GO TO 10
0211     952      CALL NUMBER(4.15,Y,.1Z1,DEPTH,0.,0)
0212              CALL PLOT(4.0,Y,3)
0213              CALL PLOT(3.,Y,2)
0214              ISIDE=1
0215     10       CONTINUE
0216              CERROR=CASING(INC)-CTOTAL
0217              CALL SYMBOL(5.5,0.5,.07,'TOTAL CASING ERROR (FT) =',0.,25)
0218              CALL NUMBER(7.1,0.5,.07,CERROR,0.,2)
0219              IF(PULLED.LT.0.01) GO TO 1001
0220              CALL SYMBOL(5.5,0.38,.07,'TOTAL CASING PULLED (FT)=',0.,25)
0221              CALL NUMBER(7.1,0.38,.07,PULLED,0.,2)
          C
          C
          C  **PERFORATION DISPLAY**
0222     1001     IF(INP.EQ.0)GO TO 20
0223              DO 20 N=1,INP,5
0224              DEPTH1=PERF(N)
0225              DEPTH2=PERF(N+1)
0226              Y1=14.-(DEPTH1-BEGIN)*SCALE
0227              Y2=14.-(DEPTH2-BEGIN)*SCALE
0228              CALL NUMBER(2.2,Y1,.1Z1,DEPTH1,0.,0)
0229              CALL PLOT(2.6,Y1,3)
0230              CALL PLOT(3.3,Y1,2)
0231              DX=-14./175.
0232              YTOP=Y1+DX
0233              DO 22 YY=YTOP,Y2,DX
0234              CALL SYMBOL(2.95,YY,.1Z1,RCAR,0.,1)
0235              CALL SYMBOL(4.02,YY,.1Z1,LCAR,0.,1)
0236     22       CONTINUE
0237              CALL NUMBER(2.2,Y2,.1Z1,DEPTH2,0.,0)
0238              CALL PLOT(2.6,Y2,3)
0239              CALL PLOT(3.3,Y2,2)
0240              YAV=Y2 + (Y1-Y2)/2.
0241              YAV1=YAV+.13
0242              IF(PERF(N+2).EQ.7.)PERF(N+2)=B27
0243              IF(PERF(N+2).EQ.9.)PERF(N+2)=B29
0244              IF(PERF(N+2).EQ.0.)PERF(N+2)=B10
0245              IF(PERF(N+2).EQ.4.)PERF(N+2)=B4
0246              IF(PERF(N+3).EQ.0.)PERF(N+3)=WDC10
0247              IF(PERF(N+3).EQ.1.)PERF(N+3)=WDC1
0248              IF(PERF(N+3).EQ.2.)PERF(N+3)=WDC2
0249              IF(PERF(N+3).EQ.3.)PERF(N+3)=WDC3
0250              IF(PERF(N+3).EQ.4.)PERF(N+3)=WDC4
0251              IF(PERF(N+3).EQ.5.)PERF(N+3)=WDC5
0252              IF(PERF(N+3).EQ.6.)PERF(N+3)=WDC6
0253              IF(PERF(N+3).EQ.7.)PERF(N+3)=WDC7
0254              IF(PERF(N+3).EQ.8.)PERF(N+3)=WDC8
0255              IF(PERF(N+3).EQ.9.)PERF(N+3)=WDC9
0256              IF(PERF(N+4).EQ.0.)PERF(N+4)=REG
0257              IF(PERF(N+4).EQ.2.)PERF(N+4)=SPIR
0258              IF(PERF(N+4).EQ.1.)PERF(N+4)=STAG
```

```
0259              N2=N+2
0260              N4=N+4
0261              NX=N2
0262              DO 26 K=1,3
0263              HEAD(K,2)=PERF(NX)
0264              NX=NX+1
0265     26       CONTINUE
0266              DO 27 K=1,3
0267              X=0.8
0268              DO 28 I=1,2
0269              CALL SYMBOL(X,YAV1,SIZ1,HEAD(K,I),0.,4)
0270              X=1.3
0271     28       CONTINUE
0272              YAV1=YAV1-0.1
0273     27       CONTINUE
0274     20       CONTINUE
         C
         C
         C   **SETUP SCREEN DISPLAY**
0275              IF(INS.EQ.0)GO TO 3001
0276              DO 30 K=1,INS,4
0277              DEPTH1=SCREEN(K)
0278              DEPTH2=SCREEN(K+1)
0279              Y1=14.-(DEPTH1-BEGIN)*SCALE
0280              Y2=14.-(DEPTH2-BEGIN)*SCALE
0281              CALL PLOT(3.6,Y1,3)
0282              CALL PLOT(4.15,Y1,2)
0283              CALL NUMBER(4.23,Y1,SIZ1,DEPTH1,0.,0)
0284              DX=-14./175.
0285              YTOP=Y1+DX
         C   **SCREEN SYMBOLS
0286              DO 35 YY=YTOP,Y2,DX
0287              XTEMP=3.11
0288              DO 32 J=1,8
0289              CALL SYMBOL(XTEMP,YY,SIZ1,4,0.,-1)
0290              XTEMP=XTEMP+.11
0291     32       CONTINUE
0292     35       CONTINUE
0293              CALL PLOT(3.6,Y2,3)
0294              CALL PLOT(4.15,Y2,2)
0295              CALL NUMBER(4.23,Y2,SIZ1,DEPTH2,0.,0)
         C   **ENTER SCREEN SIZE AND DIAMETER**
0296              YAV=Y2+(Y1-Y2)/2. + .08
0297              CALL SYMBOL(5.5,YAV,SIZ1,'SCREEN DIAMETER (FT)=',0.,21)
0298              CALL NUMBER(6.8,YAV,SIZ1,SCREEN(K+2),0.,3)
0299              SLOTSZ=SCREEN(K+3)/1000.
0300              CALL SYMBOL(5.5,YAV-.22,SIZ1,'SLOT SIZE (IN)=',0.,15)
0301              CALL NUMBER(6.8,YAV-.22,SIZ1,SLOTSZ,0.,3)
0302     30       CONTINUE
         C
         C   **COMMENTS**
         C
         C   *  CHECK TO SEE IF ARE COMMENTS*
0303     3001     IF(COMMNT(1).EQ.BLANKS)GO TO 45
0304              IF(INCM.EQ.0)GO TO 45
0305              CALL SYMBOL(4.75,12.75,SIZ1,'COMMENTS:',0.,9)
```

```
FORTRAN IV-PLUS V02-510          10:15:33     11-SEP-79          PAGE 8
WSTCAL.FTN       /TRIBLOCKS/WR

0306              IROWS=INCM/9+1
0307              KNT=0
0308              DO 43 J=1,IROWS
0309              DO 42 K=1,8
0310              KNT=KNT+1
0311              COMT(K)=COMMNT(KNT)
0312     42       CONTINUE
0313              YY=12.75-FLOAT(J)*0.1
0314              CALL SYMBOL(4.75,YY,SIZ1,COMT,0.,32)
0315     43       CONTINUE
         C
         C
         C  **PIEZOMETER SETUP**
0316     45       CALL SYMBOL(1.1,14.,SIZ1,'PIEZOMETER TUBE ID',0.,18)
0317              IF(INPE.EQ.0)GO TO 54
0318              CALL NEWPEN(1)
0319              NPZ=INPE/3
0320              TCRMT=1.0/FLOAT(NPZ+1)
0321              LETR=0
0322              X=3.0+TCRMT
0323              DO 50 K=1,INPE,3
0324              Y1=14.-(PIEZ(K)-BEGIN)*SCALE
0325              LETR=LETR+1
0326              CALL SYMBOL(X,14.09,SIZ1,DNAM(LETR),0.,1)
0327              CALL PLOT(X,14.,3)
0328              CALL PLOT(X,Y1,2)
0329              IF(PIEZ(K+1).EQ.0.)GO TO 56
0330              Y2=14.-(PIEZ(K+1)-BEGIN)*SCALE
0331              Y3=14.-(PIEZ(K+2)-BEGIN)*SCALE
0332              DO 54 J=1,15
0333              CALL SYMBOL(X-.03,Y2,SIZ1,PERFER,0.,1)
0334              Y2=Y2-.1
0335              IF(Y2.LT.Y3)GO TO 56
0336     54       CONTINUE
0337     56       CONTINUE
0338              X=X+TCRMT
0339     50       CONTINUE
         C  **DISPLAY PIEZOMETER DIAMETER
0340              CALL SYMBOL(4.75,13.8,SIZ1,'PIEZOMETER PIPE 1-1/2 IN. DIA.',0.,30)
0341              CALL SYMBOL(4.75,13.6,SIZ1,'UNLESS OTHERWISE NOTED',0.,22)
0342     98       CALL PLOT(11.0,-4.0,-3)
0343              CALL PLOTND
0344              IF (ICNTRL .NE. 1) GO TO 9991
0345              IEFN = 33
0346              CALL SETEF (IEFN,IDS)
0347     9991     CONTINUE
0348              CALL EXIT
0349              END


PROGRAM SECTIONS

 NAME        SIZE                    ATTRIBUTES

$CODE1   007472  1949                RW,I,CON,LCL
$PDATA   001053   276                RW,D,CON,LCL
```

```
0259              N2=N+2
0260              N4=N+4
0261              NX=N2
0262              DO 26 K=1,3
0263              HEAD(K,2)=PERF(NX)
0264              NX=NX+1
0265     26       CONTINUE
0266              DO 27 K=1,3
0267              X=0.8
0268              DO 28 I=1,2
0269              CALL SYMBOL(X,YAV1,SIZ1,HEAD(K,I),0.,4)
0270              X=1.3
0271     28       CONTINUE
0272              YAV1=YAV1-0.1
0273     27       CONTINUE
0274     20       CONTINUE
         C
         C
         C   **SETUP SCREEN DISPLAY**
0275              IF(INS.EQ.0)GO TO 3001
0276              DO 30 K=1,INS,4
0277              DEPTH1=SCREEN(K)
0278              DEPTH2=SCREEN(K+1)
0279              Y1=14.-(DEPTH1-BEGIN)*SCALE
0280              Y2=14.-(DEPTH2-BEGIN)*SCALE
0281              CALL PLOT(3.6,Y1,3)
0282              CALL PLOT(4.15,Y1,2)
0283              CALL NUMBER(4.23,Y1,SIZ1,DEPTH1,0.,0)
0284              DX=-14./175.
0285              YTOP=Y1+DX
         C   **SCREEN SYMBOLS
0286              DO 35 YY=YTOP,Y2,DX
0287              XTEMP=3.11
0288              DO 32 J=1,8
0289              CALL SYMBOL(XTEMP,YY,SIZ1,4,0.,-1)
0290              XTEMP=XTEMP+.11
0291     32       CONTINUE
0292     35       CONTINUE
0293              CALL PLOT(3.6,Y2,3)
0294              CALL PLOT(4.15,Y2,2)
0295              CALL NUMBER(4.23,Y2,SIZ1,DEPTH2,0.,0)
         C   **ENTER SCREEN SIZE AND DIAMETER**
0296              YAV=Y2+(Y1-Y2)/2. + .08
0297              CALL SYMBOL(5.5,YAV,SIZ1,'SCREEN DIAMETER (FT)=',0.,21)
0298              CALL NUMBER(6.8,YAV,SIZ1,SCREEN(K+2),0.,3)
0299              SLOTSZ=SCREEN(K+3)/1000.
0300              CALL SYMBOL(5.5,YAV-.22,SIZ1,'SLOT SIZE (IN)=',0.,15)
0301              CALL NUMBER(6.8,YAV-.22,SIZ1,SLOTSZ,0.,3)
0302     30       CONTINUE
         C
         C   **COMMENTS**
         C
         C   *  CHECK TO SEE IF ARE COMMENTS*
0303     3001     IF(COMMNT(1).EQ.BLANKS)GO TO 45
0304              IF(INCM.EQ.0)GO TO 45
0305              CALL SYMBOL(4.75,12.75,SIZ1,'COMMENTS:',0.,9)
```

```
FORTRAN IV-PLUS V02-510          14:15:33    11-SEP-79          PAGE 8
WSTCAL.FTN       /TRIBLOCKS/WR

0306              IROWS=INCM/9+1
0307              KNT=0
0308              DO 43 J=1,IROWS
0309              DO 42 K=1,8
0310              KNT=KNT+1
0311              COMT(K)=COMMNT(KNT)
0312     42       CONTINUE
0313              YY=12.75-FLOAT(J)*0.1
0314              CALL SYMBOL(4.75,YY,SIZ1,COMT,0.,32)
0315     43       CONTINUE
         C
         C
         C  **PIEZOMETER SETUP**
0316     45       CALL SYMBOL(1.1,14.,SIZ1,'PIEZOMETER TUBE ID',0.,18)
0317              IF(INPE.EQ.0)GO TO 54
0318              CALL NEWPEN(1)
0319              NPZ=INPE/3
0320              TCRMT=1.0/FLOAT(NPZ+1)
0321              LETR=0
0322              X=3.0+TCRMT
0323              DO 50 K=1,INPE,3
0324              Y1=14.-(PIEZ(K)-BEGIN)*SCALE
0325              LETR=LETR+1
0326              CALL SYMBOL(X,14.08,SIZ1,DNAM(LETR),0.,1)
0327              CALL PLOT(X,14.,3)
0328              CALL PLOT(X,Y1,2)
0329              IF(PIEZ(K+1).EQ.0.)GO TO 56
0330              Y2=14.-(PIEZ(K+1)-BEGIN)*SCALE
0331              Y3=14.-(PIEZ(K+2)-BEGIN)*SCALE
0332              DO 54 J=1,15
0333              CALL SYMBOL(X-.03,Y2,SIZ1,PERFER,0.,1)
0334              Y2=Y2-.1
0335              IF(Y2.LT.Y3)GO TO 56
0336     54       CONTINUE
0337     56       CONTINUE
0338              X=X+TCRMT
0339     50       CONTINUE
         C  **DISPLAY PIEZOMETER DIAMETER
0340              CALL SYMBOL(4,75,13.8,SIZ1,'PIEZOMETER PIPE 1-1/2 IN. DIA.',0.,30)
0341              CALL SYMBOL(4,75,13.6,SIZ1,'UNLESS OTHERWISE NOTED',0.,22)
0342     98       CALL PLOT(11.0,-4.0,-3)
0343              CALL PLOTND
0344              IF (ICNTRL .NE. 1) GO TO 9991
0345              IEFN = 33
0346              CALL SETEF (IEFN,IOS)
0347     9991     CONTINUE
0348              CALL EXIT
0349              END


PROGRAM SECTIONS

 NAME        SIZE                  ATTRIBUTES

$CODE1   007472  1949               RW,I,CON,LCL
$PDATA   001053   276               RW,D,CON,LCL
```

```
FORTRAN IV-PLUS V02-51D          14:15:33    11-SEP-79          PAGE 9
WSTCAL.FTN       /TR:BLOCKS/WR

$IDATA   001350    372              RW,D,CON,LCL
$VARS    007306   1A91              RW,D,CON,LCL
$TEMPS   000022      9              RW,D,CON,LCL
T1       000044     18              RW,D,OVR,GBL
T2       000024     10              RW,D,OVR,GBL
T3       000020      8              RW,D,OVR,GBL
T4       000100     32              RW,D,OVR,GBL
T5       000020      8              RW,D,OVR,GBL
HDR      001002    257              RW,D,OVR,GBL


TOTAL SPACE ALLOCATED = 022674  4830

,LP=OP:1WSTCAL
```

WSTLPR.FLX

WELL STRUCTURES (LINE PRINTER)

```
                ----------------------------------------

00001    C ***************  [351,100]WSTLPR.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID:   CIRMIS-11-ONL-2-2
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C       THIS PROGRAM RETRIEVES PERFORATION DATA
00008    C       FROM [23,33]FRMTWST.RAN AND RELAYS IT
00009    C       TO THE LINE PRINTER
00010    C
00011    C  DATA FILES:
00012    C      NAME   LUN  TYPE  ACCESS
00013    C      FILE Q--FRMTWST  1  RAN  R
00014    C
00015    C
00016    C  LOADING SEQUENCE:
00017    C       TKB>@DR0:[351,100]WSTLPR
00018    C       INS>[351,100]WSTLPR
00019    C
00020    C
00021    C  BATTELLE MEMORIAL INSTITUTE
00022    C  PACIFIC NORTHWEST LABORATORIES
00023    C  WATER & LAND RESOURCES DEPT.
00024    C
00025    C  AUTHOR(S):       DD HOSTETLER
00026    C                   DR FRIEDRICHS
00027    C
00028    C  DATE:          INITIAL VERSION   AUGUST 1974
00029    C                 CURRENT VERSION   JANUARY 1979
00030    C
00031    C
00032    C
00033          BYTE STAT,WDES
00034          INTEGER BUFF,BUF1,BUF2,BUF3,BUF4,BUF5
00035    C
00036          LOGICAL ENDCAS, CARD1, CARD2, CARD3, CARD4, CARD5
00037    C
00038    C
00039          DIMENSION INODE(255), ITRAN(255), BUFF(768)
00040          DIMENSION BUF1(18),BUF2(10),BUF3(8),BUF4(32),BUF5(8)
00041          DIMENSION PZ(4),PF(5),CM(16),CAS(8),S(4)
00042          DIMENSION SPAC(10), CUT(3,3), TCUT(3)
00043    C
00044          COMMON/T1/DIAM,CAS
00045          COMMON/T2/PF
00046          COMMON/T3/S
00047          COMMON/T4/CM
00048          COMMON/T5/PZ
00049          COMMON/HDR/ IDM(2),WDES(12),IDM1(28),IDKNT,NSTART,
00050         1            NWORD,IDM2(213),IDISC,IUNIT,ICNTRL
00051    C
00052          EQUIVALENCE (BUF1(1),DIAM),(BUF2(1),PF(1))
00053          EQUIVALENCE (BUF3(1),S(1)),(BUF4(1),CM(1))
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:19:00  PAGE  00002

00054          EQUIVALENCE (BUF5(1),PZ(1))
00055          EQUIVALENCE (WDES(1),INODE(1)),(IDM(1),ITRAN(1))
00056    C
00057          DATA CUT  /'REGU','LAR ','    ','STAG','GERE','D   ',
00058         1          'SPIR','AL  ','    '/
00059          DATA SPAC /4H 6" ,4H12" ,4H10" ,4H24" ,4H 8" ,4H36" ,
00060         1           4H 9" ,4H 5" ,4H18" ,4H16" /
00061    C
00062          TOTADD = 0.
00063          PULLED = 0.
00064          ENDCAS = .FALSE.
00065          CARD1 = .FALSE.
00066          CARD2 = .FALSE.
00067          CARD3 = .FALSE.
00068          CARD4 = .FALSE.
00069          CARD5 = .FALSE.
00070    C
00071    C***READ TRANSFER NODE***
00072    C
00073          CALL VRECSP (TASK,ITRAN,255,,,IOS)
00074    C
00075          CALL ASNLUN(1,IOISC,IUNIT)
00076          CALL OPFILE(1,'FRMTWST',1000,,IPNT1)
00077    C
00078    C     INODE(200) IS A FLAG, 5=TI TERMINAL, ANYTHING ELSE
00079    C     WILL SEND OUTPUT TO THE LINEPRINTER, DESIGNED FOR
00080    C     ACCESS BY ARMCO TERMINAL
00081    C
00082          LUN1=6
00083          IF(INODE(200).EQ.5)LUN1=5
00084    C
00085          WRITE(LUN1,601)WDES
00086    601   FORMAT(1H1,T22,'WELL STRUCTURES DOCUMENTATION      WELL =  ',
00087         112A1,/,T25,'(MEASUREMENTS IN FEET UNLESS OTHERWISE NOTED)')
00088          KNT=0
00089    C
00090    C***RETRIEVE DATA FROM BASE***
00091    C
00092          STRT=IDKNT
00093          CALL OPR(IPNT1,STRT,BUFF,768,)
00094          I=0
00095          NTOP=NSTART+NWORD-1
00096          DO 1000 J=NSTART,NTOP
00097          I=I+1
00098          BUFF(I)=BUFF(J)
00099    1000  CONTINUE
00100    C
00101    1     KNT=KNT+1
00102          IF(KNT.GT.NWORD) GO TO 99
00103          I1=BUFF(KNT)
00104    C---CHECK SECTOR COUNT
00105          SELECT (I1)
00106    C     .
00107    C----READ 1 CARDS --- CASING ADDED
00108    C     .
00109          .  (1)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:19:00  PAGE  00003

00110          .  .  DO (I=1,18)
00111          .  .  .  KNT=KNT+1
00112          .  .  .  BUF1(I)=BUFF(KNT)
00113          .  .  ...FIN
00114          .  .  IF (.NOT.CARD1)
00115          .  .  .  CARD1 = .TRUE.
00116          .  .  .  WRITE(6,610)
00117    610   .  .  .  FORMAT(//,10X,'CASING DATA:',//,T28,'LENGTHS ADDED',
00118         1,.  .  .  T56,'SUBTOTAL',T68,'DIAMETER',/,T20,
00119         2,.  .  .  '----------',T56,'--------',T68,'--------',/)
00120          .  .  ...FIN
00121          .  .  DO (J=1,8,2)
00122          .  .  .  TOTADD = TOTADD + CAS(J)
00123          .  .  .  IF (CAS(J).LT.0.) PULLED = PUSHED+CAS(J)
00124          .  .  ...FIN
00125          .  .  WRITE(6,613) (CAS(J),J=1,8,2),TOTADD,DIAM
00126    613   .  .  FORMAT(16X,4F8.2,T55,F8.2,T70,F4.2)
00127          .  ...FIN
00128    C     .
00129    C----READ 2 CARDS --- PERFORATIONS
00130    C     .
00131          .  (2)
00132          .  .  IF (.NOT.ENDCAS) WRITE-CASING-ADDED-SUMMARY
00133          .  .  DO (I=1,10)
00134          .  .  .  KNT=KNT+1
00135          .  .  .  BUF2(I)=BUFF(KNT)
00136          .  .  ...FIN
00137          .  .  IF (.NOT.CARD2)
00138          .  .  .  WRITE(6,626)
00139    626   .  .  .  FORMAT(////,T36,'PERFORATIONS',//,T20,'FROM',T31,'TO',T39,
00140         1,.  .  .  'HOLES/ROUND',T54,'SPACING',T65,'TYPE OF CUT',/,
00141         2,.  .  .  T20,'----',T31,'--',T39,'----------',T54,'-------',T65,
00142         3,.  .  .  '----------',/)
00143          .  .  .  CARD2 = .TRUE.
00144          .  .  ...FIN
00145          .  .  NHR = PF(3)
00146          .  .  SELECT (NHR)
00147          .  .  .  (0) NHR = 10
00148          .  .  .  (7) NHR = 11
00149          .  .  .  (9) NHR = 12
00150          .  .  ...FIN
00151          .  .  NSP = PF(4)
00152          .  .  WHEN (NSP.EQ.0)
00153          .  .  .  SPACE = SPAC(10)
00154          .  .  ...FIN
00155          .  .  ELSE
00156          .  .  .  SPACE = SPAC(NSP)
00157          .  .  ...FIN
00158          .  .  NTC = PF(5)+1
00159          .  .  DO (J=1,3) TCUT(J)=CUT(J,NTC)
00160          .  .  WRITE(6,629) PF(1),PF(2),NHR,SPACE,TCUT
00161    629   .  .  FORMAT(14X,2F10.2,6X,I4,11X,A4,7X,3A4)
00162          .  ...FIN
00163    C     .
00164    C----READ 3 CARDS--- SCREEN DATA
00165    C     .
```

```
              .  (3)
              .  .  IF (.NOT.ENDCAS) WRITE-CASING-ADDED-SUMMARY
              .  .  DO (I=1,8)
              .  .  .  KNT=KNT+1
              .  .  .  BUF3(I)=BUFF(KNT)
              .  .  ...FIN
              .  .  IF (.NOT.CARD3)
              .  .  .  WRITE(6,631)
      631     .  .  .  FORMAT(////,T39,'SCREENING',//,T27,'FROM',T36,'TO',T44,
             1.  .  .  'DIAMETER',T57,'SIZE',/,T27,'----',T36,'--',T44,
             2.  .  .  '--------',T57,'----',/)
              .  .  ...FIN
              .  .  WRITE(6,632) S
      632     .  .  FORMAT(25X,2F7.2,F9.2,F11.1)
              .  .  CARD3 = .TRUE.
              .  ...FIN
      C       .
      C----READ 4 CARDS --- COMMENTS
      C       .
              .  (4)
              .  .  IF (.NOT.ENDCAS) WRITE-CASING-ADDED-SUMMARY
              .  .  DO (I=1,32)
              .  .  .  KNT=KNT+1
              .  .  .  BUF4(I)=BUFF(KNT)
              .  .  ...FIN
              .  .  WRITE(LUN1,640)CM
      640     .  .  FORMAT(/15X,'COMMENTS:',4X,16A4)
              .  ...FIN
      C       .
      C----READ 5 CARDS ---   PIEZOMETER TUBES
      C       .
              .  (5)
              .  .  IF (.NOT.ENDCAS) WRITE-CASING-ADDED-SUMMARY
              .  .  DO (I=1,8)
              .  .  .  KNT=KNT+1
              .  .  .  BUF5(I)=BUFF(KNT)
              .  .  ...FIN
              .  .  IF (.NOT.CARD5)
              .  .  .  WRITE(6,650)
      650     .  .  .  FORMAT(//////,10X,'PIEZOMETER TUBES:',//,T38,'PERFORATIONS',/,
             1.  .  .  T17,'TUBE',T26,'LENGTH',T39,'FROM',T47,'TO',/,
             2.  .  .  T17,'----',T26,'------',T39,'----',T47,'--')
              .  .  .  CARD5 = .TRUE.
              .  .  ...FIN
              .  .  WRITE(6,651) PZ
      651     .  .  FORMAT(T18,A1,T26,F5.0,T38,F5.0,T45,F5.0)
              .  ...FIN
              ...FIN
              GO TO 1
      C
      99      IF (ICNTRL .NE. 1) GO TO 9991
              IEFN = 33
              CALL SETEF (IEFN,IOS)
      9991    CALL EXIT

      ------------------------------------------
```

```
00220          TO WRITE-CASING-ADDED-SUMMARY
00221          .  ENDCAS = .TRUE.
00222          .  HNDTOT = CAS(2)
00223          .  DO (J=2,8,2)
00224          .  .  IF (CAS(J).NE.0.) HNDTOT=CAS(J)
00225          .  ...FIN
00226          .  TOTERR = TOTADD-HNDTOT
00227          .  WRITE(6,620) TOTADD,HNDTOT,TOTERR,PULLED
00228    620   .  FORMAT(/,T48,'TOTAL =',F8.2,//,T32,'HAND CALCULATED TOTAL =',
00229         1.  F8.2,/,T32,'TOTAL ERROR IN LENGTH =',F8.2,/,
00230         2.  T32,'TOTAL CASING PULLED   =',F8.2)
00231          ...FIN
00232          END
```

PROCEDURE CROSS-REFERENCE TABLE

00220 WRITE-CASING-ADDED-SUMMARY
00132 00167 00186 00198

(FLECS VERSION 22.46)

```
00233          SUBROUTINE READER(BUFF,IDKNT,KNT,IPNT1)
00234          INTEGER BUFF
00235          DIMENSION BUFF(256)
00236          STRT=IDKNT-1
00237          CALL DPR(IPNT1,STRT,BUFF,256,)
00238          IDKNT=IDKNT+1
00239          KNT=0
00240          RETURN
00241          END
```

(FLECS VERSION 22.46)

WLGCAL.FLX

DRILLERS LOG (CALCOMP)

```
00001   C ***************  [351,100]WLGCAL.FLX  ***************
00002   C
00003   C
00004   C  PROGRAM ID:  CIRMIS-11-ONL-3-2     "HANFORD"
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C       THIS PROGRAM OUTPUTS DRILLERS LOGS TO LINE PRINTER.
00008   C       DATA OUTPUT: SOIL TYPES, DRILLERS EXPLANATIONS, AND COMMENTS.
00009   C
00010   C  DATA FILES:
00011   C       NAME                    LUN             TYPE            ACCESS
00012   C       FILE Q--FRMTWLG          1               RAN               R
00013   C
00014   C
00015   C       MCR>TKB @[351,100]WLGCAL
00016   C       MCR>INS [351,100]WLGCAL
00017   C
00018   C
00019   C  BATTELLE MEMORIAL INSTITUTE
00020   C  PACIFIC NORTHWEST LABORATORIES
00021   C  WATER & LAND RESOURCES DEPT.
00022   C
00023   C  AUTHOR(S):     DR FRIEDRICHS
00024   C
00025   C  DATE:          INITIAL VERSION   JANUARY 1977
00026   C                 CURRENT VERSION   OCTOBER 1978
00027   C
00028   C
00029   C
00030         BYTE DAT(9),TIM(8),STRNG1(720),STRNG2(100),BLANK
00031         BYTE WLDES
00032         LOGICAL FIRST, DONE
00033   C
00034         DIMENSION SOILS(180),EXPL(180),STRING(180),SLAST(180)
00035         DIMENSION ELAST(100),COMM(500),IBUF(253)
00036         DIMENSION NDUM(3000),NBUF(256),INODE(255)
00037         DIMENSION DDATE(2),T(2),TC(2)
00038   C
00039         COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(21),
00040        1            IBLK,NSTRT,NWRD,IDM3(177),OWN(6),DRL(6),IDM4(7),
00041        2            IDISC,IUNIT,ICNTRL
00042         COMMON/BLK2/ BUF(1500)
00043   C
00044         EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
00045         EQUIVALENCE (BUF(1),NDUM(1)), (STRING(1),STRNG1(1))
00046   C
00047         DATA BLANKS /4H    /
00048         DATA BLANK  /1H /
00049   C
00050   C
00051   C
00052   C *** READ THE TRANSFER NODE ***
00053   C
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:19:14  PAGE  00002

00054   C       TASK  =  RAD50('MNTR45')
00055           CALL VRECSP (TASK,INODE,255,,,IDS)
00056   C
00057           CALL ASNLUN(1,IDISC,IUNIT)
00058           CALL OPFILE(1,'FRMTWLG',8000,,IPNT1)
00059   C
00060   C
00061   C
00062   C
00063           RETRIEVE-DATA
00064           PLOT-CASING-AND-LABELS
00065           NS     = 0
00066           NEX    = 0
00067           ITYPE  = 0
00068           PREVD  = 0.0
00069           FIRST  = .TRUE.
00070           DONE   = .FALSE.
00071   C
00072   C
00073   C  ****** READ DATA FROM BUFFER ******
00074   C
00075           REPEAT UNTIL (DONE)
00076           .  N      = N+1
00077           .  I1     = BUF(N)
00078           .  SELECT (I1)
00079   C       .  .
00080   C  CARD TYPE 1---HEADER INFO
00081           .  .  (1)
00082           .  .  .  IF (ITYPE. NE .0)
00083           .  .  .  .  SELECT (ITYPE)
00084           .  .  .  .  .  (1) PLOT-HEADER
00085           .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
00086           .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
00087           .  .  .  .  .  (4) PLOT-LAST-COMMENT
00088           .  .  .  .  ...FIN
00089           .  .  .  ...FIN
00090           .  .  .  ITYPE = 1
00091           .  .  ...FIN
00092   C       .  .
00093   C  CARD TYPE 2---STRATIFICATION
00094           .  .  (2)
00095           .  .  .  IF (ITYPE. NE .0)
00096           .  .  .  .  SELECT (ITYPE)
00097           .  .  .  .  .  (1) PLOT-HEADER
00098           .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
00099           .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
00100           .  .  .  .  .  (4) PLOT-LAST-COMMENT
00101           .  .  .  .  ...FIN
00102           .  .  .  ...FIN
00103           .  .  .  ITYPE = 2
00104           .  .  .  N      = N+1
00105           .  .  .  TIM2   = BUF(N)
00106           .  .  .  N      = N+1
00107           .  .  .  DEPTH  = BUF(N)
00108           .  .  .  N      = N+1
00109           .  .  .  ISAMP  = BUF(N)
```

```
00110          .  .  .  N      = N+1
00111          .  .  .  ICONT  = BUF(N)
00112          .  .  .  N      = N+1
00113          .  .  .  NEXPL  = BUF(N)
00114          .  .  .  IF (NEXPL.NE.0)
00115          .  .  .  .  DO (J = 1,NEXPL)
00116          .  .  .  .  .  N      = N+1
00117          .  .  .  .  .  EXPL(J) = BUF(N)
00118          .  .  .  .  ...FIN
00119          .  .  .  ...FIN
00120          .  .  ...FIN
00121    C     .  .
00122    C  CARD TYPE 3---SOILS
00123          .  .  (3)
00124          .  .  .  IF (ITYPE. NE .0)
00125          .  .  .  .  SELECT (ITYPE)
00126          .  .  .  .  .  (1) PLOT-HEADER
00127          .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
00128          .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
00129          .  .  .  .  .  (4) PLOT-LAST-COMMENT
00130          .  .  .  .  ...FIN
00131          .  .  .  ...FIN
00132          .  .  .  ITYPE = 3
00133          .  .  .  N     = N+1
00134          .  .  .  TIM3  = BUF(N)
00135          .  .  .  N     = N+1
00136          .  .  .  DEPTH3 = BUF(N)
00137          .  .  .  N     = N+1
00138          .  .  .  NSOIL = BUF(N)
00139          .  .  .  IF (NSOIL.NE.0)
00140          .  .  .  .  DO (J = 1,NSOIL)
00141          .  .  .  .  .  N     = N+1
00142          .  .  .  .  .  SOILS(J) = BUF(N)
00143          .  .  .  .  ...FIN
00144          .  .  .  ...FIN
00145          .  .  ...FIN
00146    C     .  .
00147    C  CARD TYPE 4---COMMENTS
00148          .  .  (4)
00149          .  .  .  IF (ITYPE. NE .0)
00150          .  .  .  .  SELECT (ITYPE)
00151          .  .  .  .  .  (1) PLOT-HEADER
00152          .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
00153          .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
00154          .  .  .  .  .  (4) PLOT-LAST-COMMENT
00155          .  .  .  .  ...FIN
00156          .  .  .  ...FIN
00157          .  .  .  ITYPE = 4
00158          .  .  .  N     = N+1
00159          .  .  .  TC(1) = BUF(N)
00160          .  .  .  N     = N+1
00161          .  .  .  TC(2) = BUF(N)
00162          .  .  .  N     = N+1
00163          .  .  .  CDEP  = BUF(N)
00164          .  .  .  N     = N+1
00165          .  .  .  NCOM  = BUF(N)
```

```
00166          .  .  .  IF (NCOM .NE. 0)
00167          .  .  .  .  DO (J = 1,NCOM)
00168          .  .  .  .  .  N      = N+1
00169          .  .  .  .  .  COMM(J) = BUF(N)
00170          .  .  .  .  ...FIN
00171          .  .  .  ...FIN
00172          .  .  ...FIN
00173    C     .  .
00174    C  CARD TYPE 5---END OF DATA (PLOT REMAINING DATA IN BUFFER)
00175          .  .  (5)
00176          .  .  .  IF (ITYPE. NE .0)
00177          .  .  .  .  SELECT (ITYPE)
00178          .  .  .  .  .  (1) PLOT-HEADER
00179          .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
00180          .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
00181          .  .  .  .  .  (4) PLOT-LAST-COMMENT
00182          .  .  .  .  ...FIN
00183          .  .  .  ...FIN
00184          .  .  .  ITYPE = 5
00185          .  .  .  DONE = .TRUE.
00186          .  .  ...FIN
00187          .  ...FIN
00188          ...FIN
00189          NEXPL = 0
00190          DEPTH = TOTDEP
00191          PLOT-LAST-STRATIGRAPHY
00192          CALL PLOT (BOT+4.,0,0,3)
00193          CALL PLOTND
00194          IF (ICNTRL .EQ. 1)
00195          .  IEFN = 33
00196          .  CALL SETEF(IEFN,IDS)
00197          ...FIN
00198          CALL EXIT

       ----------------------------------------

00199          TO RETRIEVE-DATA
00200          .  NWORD = NWRD
00201          .  KBLK  = IBLK
00202          .  JB    = NSTRT
00203          .  STRT  = KBLK
00204          .  CALL DPR(IPNT1,STRT,NBUF,256.)
00205          .  DO (J  =  1,NWORD)
00206          .  .  JB = JB+1
00207          .  .  NDUM(J) = NBUF(JB)
00208          .  .  IF (JB. GE .256)
00209          .  .  .  KBLK = KBLK+1
00210          .  .  .  STRT = KBLK
00211          .  .  .  CALL DPR(IPNT1,STRT,NBUF,256.)
00212          .  .  .  JB = 0
00213          .  .  ...FIN
00214          .  ...FIN
00215    C     .
00216          .  NTOT     = BUF(4)
00217          .  DDATE(1) = BUF(5)
00218          .  DDATE(2) = BUF(6)
```

```
00219          .  TOTDEP   = BUF(7)
00220          .  N = 7
00221          ...FIN

       ----------------------------------------

00222          TO PLOT-CASING-AND-LABELS
00223    C     .
00224          .  CALL PLOTS(0.,0.,7)
00225          .  CALL NEWPEN(1)
00226          .  CALL FACTOR(1.0)
00227          .  CALL PLOT(0.,0.,-3)
00228    5     .  CALL TIME(TIM)
00229          .  CALL DATE(DAT)
00230          .  X = 0.
00231          .  Y = 1.
00232          .  SIZ = 0.15
00233          .  CALL SYMBOL(X,Y,SIZ,'WELL NO.',90.,8)
00234    C     PLOT WELL DESIGNATION
00235          .  CALL SYMBOL(X,Y+1.3,SIZ,WLDES,90.,12)
00236          .  SIZ = 0.08
00237          .  CALL SYMBOL(X,Y+3.2,SIZ,DAT,90.,9)
00238          .  CALL SYMBOL(X,Y+4.1,SIZ,TIM,90.,8)
00239          .  X = .3
00240          .  SIZ = 0.045
00241          .  CALL NEWPEN(1)
00242          .  CALL SYMBOL(X,0.0,SIZ,'DATE:',90.,5)
00243          .  CALL SYMBOL(X+.1,0.0,SIZ,'CASING:',90.,7)
00244          .  CALL SYMBOL(X+.2,0.0,SIZ,'DRILLER:',90.,8)
00245          .  CALL SYMBOL(X+.3,0.0,SIZ,'FORMAN:',90.,6)
00246          .  CALL NEWPEN(1)
00247          .  CALL PLOT(1.0,0.0,-3)
00248          .  SIZ = 0.09
00249          .  X = -.1
00250          .  CALL SYMBOL(X,0.0,SIZ,'STRATIGRAPHY',90.,12)
00251          .  CALL SYMBOL(X,3.2,SIZ,'MATERIALS PENETRATED',90.,20)
00252          .  CALL SYMBOL(X,6.0,SIZ,'COMMENTS',90.,8)
00253          .  CALL PLOT(0.0,7.5,3)
00254          .  CALL PLOT(0.0,2.64,2)
00255          .  CALL PLOT(0.0,2.00,3)
00256          .  CALL PLOT(0.0,0.00,2)
00257    C  SET SCALE AT 20FT/IN
00258          .  DSCALE = 20.
00259          .  BOT = TOTDEP/DSCALE
00260          .  CALL PLOT(BOT,2.0,3)
00261          .  CALL PLOT(-.3,2.0,2)
00262          .  CALL PLOT(-.3,2.04,2)
00263          .  CALL PLOT(BOT,2.04,2)
00264          .  CALL PLOT(BOT,2.6,3)
00265          .  CALL PLOT(-.3,2.6,2)
00266          .  CALL PLOT(-.3,2.64,2)
00267          .  CALL PLOT(BOT,2.64,2)
00268          ...FIN

       ----------------------------------------
```

```
00269          TO PLOT-HEADER
00270    C   CARD TYPE 1
00271    C
00272          .  IF (FIRST)
00273          .  .  FIRST = .FALSE.
00274          .  .  SIZ = 0.045
00275          .  .  X = -.7
00276          .  .  CALL SYMBOL(X,0.4,SIZ,ODATE,90.,6)
00277          .  .  CALL NUMBER(X+.1,0.4,SIZ,CASELV,90.,1)
00278          .  .  DO (J = 1,6) STRING(J) = ORL(J)
00279          .  .  CALL SYMBOL(X+.2,0.4,SIZ,STRING,90.,24)
00280          .  .  DO (J = 1,6) STRING(J) = OWN(J)
00281          .  .  CALL SYMBOL(X+.3,0.4,SIZ,STRING,90.,18)
00282          .  ...FIN
00283          ...FIN

       ----------------------------------------

00284          TO PLOT-LAST-STRATIGRAPHY
00285    C   CARD TYPE 2
00286    C
00287          .  IF(NEX.EQ.0) GO TO 74
00288          .  IF(NEX.NE.NEXPL) GO TO 72
00289          .  DO (J = 1,NEX)
00290          .  .  IF(ELAST(J).NE.EXPL(J)) GO TO 72
00291          .  ...FIN
00292    C  SAME
00293          .  GO TO 77
00294    C  NEW STRATA--PLOT LAST ONE
00295    72    .  X  = (DEPTHE+DEPTH)/(2.*DSCALE)
00296          .  XT = (DEPTHE+.7)/DSCALE
00297          .  XB = (DEPTH-.7)/DSCALE
00298          .  CALL PLOT(XT,1.9,3)
00299          .  CALL PLOT(XT,1.6,2)
00300          .  CALL PLOT(X,1.6,2)
00301          .  CALL PLOT(X,1.25,2)
00302          .  CALL PLOT(X,1.6,2)
00303          .  CALL PLOT(XB,1.6,2)
00304          .  CALL PLOT(XB,1.9,2)
00305          .  LE = 0
00306          .  DO (J = 1,NEX)
00307          .  .  IF (ELAST(J) .NE. BLANKS)
00308          .  .  .  LE = LE+1
00309          .  .  .  STRING(J) = ELAST(J)
00310          .  .  ...FIN
00311          .  ...FIN
00312    74    .  LEN = LE*4
00313          .  CALL SYMBOL(X,0.9,SIZ,STRING,90.,LEN)
00314          .  YY = SIZ*1.5+FLOAT(LEN)/2.
00315          .  XX = X+SIZ/3.
00316          .  CALL PLOT(XX,0.,3)
00317          .  CALL PLOT(XX,YY,2)
00318          .  IF (NEXPL .NE. 0)
00319          .  .  DO (J = 1,NEXPL)
00320          .  .  .  ELAST(J) = EXPL(J)
00321          .  .  ...FIN
```

```
00322          .  ...FIN
00323          .  NEX = NEXPL
00324    77    .  DEPTHE = DEPTH
00325          ...FIN
```

```
00326          TO PLOT-LAST-SOIL-TYPE
00327    C     .
00328    C   CARD TYPE 3
00329    C     .
00330    C  CHECK FOR SOIL
00331    80    .  IF(NS.EQ.0) GO TO 84
00332          .  IF(NS.NE.NSOIL) GO TO 82
00333          .  DO (J = 1,NSOIL)
00334          .  .  IF(SLAST(J).NE.SOILS(J)) GO TO 82
00335          .  ...FIN
00336    C   SAME SOIL
00337          .  TIMEL = TIM2
00338          .  GO TO 87
00339    C  NEW SOIL--PLOT LAST ONE
00340    82    .  X = DEPTHL/DSCALE
00341          .  CALL PLOT(X,2.0,3)
00342          .  CALL PLOT(X,2.64,2)
00343          .  CALL NUMBER(X,2.75,SIZ,DEPTHL,90.,1)
00344          .  X = (PREVD+DEPTHL)/(2.*DSCALE)
00345          .  LS = 0
00346          .  DO (J = 1,NS)
00347          .  .  IF (SLAST(J) .NE. BLANKS)
00348          .  .  .  LS = LS+1
00349          .  .  .  STRING(LS) = SLAST(J)
00350          .  .  ...FIN
00351          .  ...FIN
00352          .  LEN = LS*4
00353          .  CALL SYMBOL(X,3.1,SIZ,STRING,90.,LEN)
00354          .  PREVD = DEPTHL
00355    84    .  CONTINUE
00356          .  IF (NSOIL .NE. 0)
00357          .  .  DO (J=1,NSOIL)
00358          .  .  .  SLAST(J) = SOILS(J)
00359          .  .  ...FIN
00360          .  ...FIN
00361          .  NS = NSOIL
00362    87    .  DEPTHL = DEPTH3
00363          ...FIN
```

```
00364          TO PLOT-LAST-COMMENT
00365    C     .
00366    C  CARD TYPE 4
00367    C     .
00368    C   CHECK FOR COMMENT
00369          .  X  = COMP/DSCALE
00370          .  LC = 0
00371          .  IF (NCOM .GT. 0)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:19:14  PAGE  00008

  00372          .  .  DO (J = 1,NCOM)
  00373          .  .  .  IF (NCOM .NE. BLANKS)
  00374          .  .  .  .  LC = LC+1
  00375          .  .  .  .  STRING(J) = COMM(J)
  00376          .  .  .  ...FIN
  00377          .  .  ...FIN
  00378          .  .  LEN = LC*4
  00379          .  .  NCH = 0
  00380          .  .  DO (J = 1,LEN)
  00381          .  .  .  NCH = NCH+1
  00382          .  .  .  STRNG2(NCH) = STRNG1(J)
  00383          .  .  .  IF (NCH .GE. 80)
  00384          .  .  .  .  IF (STRNG2(NCH) .EQ. BLANK)
  00385    C     WRITE THIS LINE
  00386          .  .  .  .  .  CALL SYMBOL(X,.6,.SIZ,STRNG2,90.,NCH)
  00387          .  .  .  .  .  NCH = 0
  00388          .  .  .  .  .  X = X+.1
  00389          .  .  .  .  ...FIN
  00390          .  .  .  ...FIN
  00391          .  .  ...FIN
  00392          .  .  IF (NCH .LT. 80) CALL SYMBOL(X,.6,.SIZ,STRNG2,90.,NCH)
  00393          .  ...FIN
  00394          ...FIN
  00395          END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00222 PLOT-CASING-AND-LABELS
                00064

          00269 PLOT-HEADER
                00084  00097  00126  00151  00178

          00364 PLOT-LAST-COMMENT
                00087  00100  00129  00154  00181

          00326 PLOT-LAST-SOIL-TYPE
                00086  00099  00128  00153  00180

          00284 PLOT-LAST-STRATIGRAPHY
                00085  00098  00127  00152  00179  00191

          00199 RETRIEVE-DATA
                00063


          (FLECS VERSION 22.46)

          ----------------------------------------
```

WLGLPR.FLX

DRILLERS LOG (LINE PRINTER)

```
00001    C ***************  [351,100]WLGLPR.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID:  CIRMIS-11-ONL-3-2     "HANFORD"
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C       THIS PROGRAM OUTPUTS DRILLERS LOGS TO LINE PRINTER.
00008    C       DATA OUTPUT: SOIL TYPES, DRILLERS EXPLANATIONS, AND COMMENTS.
00009    C
00010    C  DATA FILES:
00011    C       NAME                    LUN             TYPE            ACCESS
00012    C       FILE D--FRMTWLG          1               RAN               R
00013    C
00014    C
00015    C       MCR>TKB *[351,100]WLGLPR
00016    C       MCR>INS [351,100]WLGLPR
00017    C
00018    C
00019    C  BATTELLE MEMORIAL INSTITUTE
00020    C  PACIFIC NORTHWEST LABORATORIES
00021    C  WATER & LAND RESOURCES DEPT.
00022    C
00023    C  AUTHOR(S):     DR FRIEDRICHS
00024    C
00025    C  DATE:          INITIAL VERSION   JANUARY 1977
00026    C                 CURRENT VERSION   OCTOBER 1978
00027    C
00028    C
00029    C
00030          BYTE DAT(9),TIM(8),STRNG1(720),STRNG2(100),BLANK
00031          BYTE WLDES
00032    C
00033          DIMENSION SOILS(180),EXPL(180),STRING(180),SLAST(180)
00034          DIMENSION D(5),C(6),COMM(500),IBUF(255)
00035          DIMENSION NDUM(3000),NBUF(256),INODE(255)
00036          DIMENSION DDATE(2),T(2),TC(2)
00037    C
00038          COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(21),
00039         1            IBLK,NSTRT,NWRD,IDM3(177),OWN(6),DRL(6),IDM4(7),
00040         2            IDISC,IUNIT,ICNTRL
00041          COMMON/BLK2/ BUF(1500)
00042    C
00043          EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
00044          EQUIVALENCE (BUF(1),NDUM(1)), (STRING(1),STRNG1(1))
00045    C
00046          DATA BLANKS /4H    /
00047          DATA BLANK  /1H /
00048    C
00049    C
00050    C
00051    C  READ TRANSFER NODE
00052    C     TASK = RAD50('MNTR45')
00053          CALL VRELSP (TASK,INODE,255,,,IDS)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14144108  PAGE  00002

 00054          JPNT = 0
 00055          NDEL = 0
 00056    C
 00057    C
 00058          CALL ASNLUN(1,IDISC,IUNIT,0)
 00059          CALL OPFILE(1,'FRMTWLG',8000,,IPNT1)
 00060    C
 00061    C
 00062    C
 00063          NWORD = NWRD
 00064          KBLK = IBLK
 00065          JB = NSTRT
 00066          STRT = KBLK
 00067          CALL OPR(IPNT1,STRT,NBUF,256,)
 00068          DO (J = 1,NWORD)
 00069          .  JB = JB+1
 00070          .  NDUM(J) = NBUF(JB)
 00071          .  IF (JB. GE .256)
 00072          .  .  KBLK = KBLK+1
 00073          .  .  STRT = KBLK
 00074          .  .  CALL OPR(IPNT1,STRT,NBUF,256,)
 00075          .  .  JB = 0
 00076          .  ...FIN
 00077          ...FIN
 00078    C
 00079          NTOT      = BUF(4)
 00080          DDATE(1)  = BUF(5)
 00081          DDATE(2)  = BUF(6)
 00082          TOTDEP    = BUF(7)
 00083          N = 7
 00084          WRITE(5,505) WLDES,TOTDEP
 00085    505   FORMAT(1H1,20X,'WELL NO. ',12A1,5X,'TOTAL DEPTH=',F10.2,///)
 00086    C
 00087    C
 00088          ND     = 0
 00089          NS     = 0
 00090          NEX    = 0
 00091          ITYPE = 0
 00092    C
 00093    C
 00094    C  ****** READ DATA FROM BUFFER ******
 00095    C
 00096    10    N = N+1
 00097          IF(N.GT.NTOT) GO TO 98
 00098          I1 = BUF(N)
 00099          SELECT (I1)
 00100    C  CARD TYPE 1---HEADER INFO
 00101          .  (1)
 00102          .  .  ITYPE = 1
 00103          .  .  WRITE(5,514) DDATE,DRL,OWN
 00104    514   .  .  FORMAT(' DATE:  ',A4,A2,/,' DRILLER:  ',6A4,/' FORMAN:    ',6A4,
 00105         1.  .  //)
 00106          .  ...FIN
 00107    C     .
 00108    C  CARD TYPE 2---STRATIFICATION
 00109          .  (2)
```

```
00110          .  .  N = N+1
00111          .  .  TIM2 = BUF(N)
00112          .  .  N = N+1
00113          .  .  DEPTH = BUF(N)
00114          .  .  N = N+1
00115          .  .  ISAMP = BUF(N)
00116          .  .  N = N+1
00117          .  .  ICONT = BUF(N)
00118          .  .  N = N+1
00119          .  .  NEXPL = BUF(N)
00120          .  .  IF (NEXPL. NE .0)
00121          .  .  .  DO (J = 1,NEXPL)
00122          .  .  .  .  N = N+1
00123          .  .  .  .  EXPL(J) = BUF(N)
00124          .  .  .  ...FIN
00125          .  .  .  WRITE(5,516) (EXPL(J),J = 1,NEXPL)
00126    516   .  .  .  FORMAT(//,2X,29A4)
00127          .  .  .  WRITE(5,5170)
00128    5170  .  .  .  FORMAT(/,5X,'DEPTH',T20,'MATERIALS PENETRATED',/)
00129          .  .  ...FIN
00130          .  ...FIN
00131    C     .
00132    C  CARD TYPE 3---SOILS
00133          .  (3)
00134          .  .  ITYPE = 3
00135          .  .  N = N+1
00136          .  .  TIM3 = BUF(N)
00137          .  .  N = N+1
00138          .  .  DEPTH3 = BUF(N)
00139          .  .  N = N+1
00140          .  .  NSOIL = BUF(N)
00141          .  .  IF (NSOIL. NE .0)
00142          .  .  .  DO (J = 1,NSOIL)
00143          .  .  .  .  N = N+1
00144          .  .  .  .  SOILS(J) = BUF(N)
00145          .  .  .  ...FIN
00146          .  .  .  WRITE(5,517) DEPTH3,(SOILS(J),J = 1,NSOIL)
00147    517   .  .  .  FORMAT(4X,F7.0,5X,(29A4))
00148          .  .  ...FIN
00149          .  ...FIN
00150    C     .
00151    C  CARD TYPE 4---COMMENTS
00152          .  (4)
00153          .  .  ITYPE = 4
00154          .  .  N = N+1
00155          .  .  TC(1) = BUF(N)
00156          .  .  N = N+1
00157          .  .  TC(2) = BUF(N)
00158          .  .  N = N+1
00159          .  .  CDEP = BUF(N)
00160          .  .  N = N+1
00161          .  .  NCOM = BUF(N)
00162          .  .  IF (NCOM. NE .0)
00163          .  .  .  DO (J = 1,NCOM)
00164          .  .  .  .  N = N+1
00165          .  .  .  .  COMM(J) = BUF(N)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14144108  PAGE  00004

 00166          .  .  .  ...FIN
 00167          .  .  .  WRITE(3,560) CDEP,(COMM(J),J = 1,NCOM)
 00168   560    .  .  .  FORMAT(4X,F7.0,5X,'**COMMENT**  ',20A4)
 00169          .  .  ...FIN
 00170          .  ...FIN
 00171   C      .
 00172   C  CARD TYPE 5---READ THRU END OF FILE
 00173          .  (5)
 00174          .  .  IF (ICNTRL. EQ .1)
 00175          .  .  .  IEFN = 33
 00176          .  .  .  CALL SETEF(IEFN,IDS)
 00177          .  .  ...FIN
 00178          .  .  IF (I1. EQ .5) CALL EXIT
 00179          .  ...FIN
 00180          ...FIN
 00181          GO TO 10
 00182   C
 00183   C
 00184   C  ERROR IN DATA LENGTH
 00185   98     WRITE(6,698)
 00186   698    FORMAT(' ERROR IN DATA LENGTH')
 00187          IF (ICNTRL. EQ .1)
 00188          .  IEFN = 33
 00189          .  CALL SETEF(IEFN,IDS)
 00190          ...FIN
 00191          STOP
 00192          END

         (FLECS VERSION 22.46)

         ----------------------------------------
```

WLGGLD.FTN

DRILLERS LOG (GOULD)

```
C ***************  DR0:[351,100]WLGGLD.FTN  ***************
C
C
C  PROGRAM ID:  SRV-11-WEL-3-3
C
C  PROGRAM DESCRIPTION:
C       THIS PROGRAM OUTPUTS DRILLERS LOGS TO GOULD PLOTTER,
C       DATA OUTPUT: STRATIFICATION, SOIL TYPES, AND COMMENTS,
C       WELL SCALE = 20 FEET PER INCH.  THIS VERSION DOESN'T SHADE,
C               "HANFORD RESERVATION"
C
C  DATA FILES:
C       NAME                    LUN             TYPE            ACCESS
C       FILE G--FRMTWLG         1               RAN             R
C       GOULD OUTPUT            7                               W
C
C
C       MCR>TKB @DB0:[351,100]WLGGLD
C
C
C  BATTELLE MEMORIAL INSTITUTE
C  PACIFIC NORTHWEST LABORATORIES
C  WATER & LAND RESOURCES DEPT.
C
C  AUTHOR(S):    DR FRIEDRICHS
C
C
C  DATE:         INITIAL VERSION   JANUARY 1977
C                CURRENT VERSION   APRIL   1978
C
C
C
0001            BYTE DAT(9),TIM(8),STRNG1(720),STRNG2(100),BLANK,D(48)
0002            BYTE A(6),WLDES
      C
0003            INTEGER BUFFER(4096),X,Y,XX,YY,SIZ,BOT,XT,XB
      C
0004            DIMENSION SOILS(180),STRING(180),SLAST(180)
0005            DIMENSION EXPL(100),ELAST(100),COMM(200),INODE(253)
0006            DIMENSION NDUM(3000),NBUF(256),IBUF(255)
0007            DIMENSION DDATE(2),T(2),TC(2)
      C
0008            COMMON/HDR/ IDM(2),WLDES(12),IDUM1(10),CASE,IDUM2(21),IBLK,
                1           NSTRT,NWRD,IDUM3(177),OWN(6),DRL(6),IDM4(7),IDISC,
                2           IUNIT,ICNTRL
0009            COMMON/BLK2/ BUF(1500)
      C
0010            EQUIVALENCE (BUF(1),NDUM(1))
0011            EQUIVALENCE (STRING(1),STRNG1(1))
0012            EQUIVALENCE (INODE(1),WLDES(1)),(INODE(1),IBUF(3))
      C
0013            DATA BLANKS /4H    /
0014            DATA BLANK  /1H /
      C
      C
      C
```

```
        C  READ TRANSFER NODE
        C       TASK = RAD50('MNTR45')
0015            CALL VRECSP (TASK,IBUF,255,,,IDS)
        C
0016            CALL ASNLUN(1,IDISC,IUNIT)
0017            CALL DPFILE(1,'FRMTWLG',8000,,IPNT1)
        C
0018            NWORD=NWRD
0019            KBLK=IBLK
0020            JR=NSTRT
0021            STRT=KBLK
0022            CALL DPR(IPNT1,STRT,NBUF,256,)
0023            DO 6 J=1,NWORD
0024            JR=JR+1
0025            NDUM(J)=NBUF(JR)
0026            IF(JR.LT.256) GO TO 6
0027            KBLK=KBLK+1
0028            STRT=KBLK
0029            CALL DPR(IPNT1,STRT,NBUF,256,)
0030            JR=0
0031    6       CONTINUE
        C
0032            NTOT=BUF(4)
0033            DDATE(1)=BUF(5)
0034            DDATE(2)=BUF(6)
0035            TOTDEP=BUF(7)
        C  SET SCALE AT 20FT/IN
0036            DSCALE=20./100.
0037            BOT=TOTDEP/DSCALE + 200
0038            NN=TOTDEP/20.
0039            NN=NN*100 + 200
0040            NN=(NN/32+2)*32
        C
        C
0041            CALL TIME(TIM)
0042            CALL DATE(DAT)
0043            CALL DOTINI(7,BUFFER,32,1024)
0044            DO 9990 INDEX=1,NN,32
0045            X=100
0046            Y=100
0047            SIZ=24
0048            CALL DOTCHR(X,Y,'WELL NO.',,SIZ,1)
        C   PLOT WELL DESIGNATION
0049            CALL DOTCHR(X,230,WLDES,12,SIZ,1)
0050            X=190
0051            SIZ=16
0052            CALL DOTCHR(X,8,'STRATIGRAPHY',,SIZ,1)
0053            CALL DOTCHR(X,300,'MATERIALS PENETRATED',,SIZ,1)
0054            CALL DOTCHR(X,660,'COMMENTS',,SIZ,1)
0055            X=100
0056            SIZ=8
0057            CALL DOTCHR(X,520,DAT,9,SIZ,1)
0058            CALL DOTCHR(X,610,TIM,8,SIZ,1)
0059            X=200
0060            CALL DOTLIN(X,750,X,264)
0061            CALL DOTLIN(X,200,X,0)
```

```
        C
0062            N=7
0063            CALL DOTLIN(BOT,200,170,200)
0064            CALL DOTLIN(170,200,170,204)
0065            CALL DOTLIN(170,204,BOT,204)
0066            CALL DOTLIN(BOT,260,170,260)
0067            CALL DOTLIN(170,260,170,264)
0068            CALL DOTLIN(170,264,BOT,264)
        C
        C
0069            PREVD=0.0
0070            NS=0
0071            NEX=0
0072            ITYPE=0
0073            NCARD1=0
0074            SIZ=8
        C
        C
        C  ****** READ DATA FROM BUFFER ******
        C
0075    10      N=N+1
0076            IF(N.GT.NTOT) GO TO 98
0077            I1=BUF(N)
0078            GO TO(12,15,30,50,99),I1
0079    12      IF(ITYPE.EQ.0) GO TO 13
0080            IPLACE=1
0081            GO TO(60,70,80,90),ITYPE
        C
        C  CARD TYPE 1---HEADER INFO
0082    13      ITYPE=1
0083             GO TO 10
        C  CARD TYPE 2---STRATIFICATION
0084    15      IF(ITYPE.EQ.0) GO TO 16
0085            IPLACE=2
0086            GO TO (60,70,80,90),ITYPE
0087    16      ITYPE=2
0088            N=N+1
0089            TIM2=BUF(N)
0090            N=N+1
0091            DEPTH=BUF(N)
0092            N=N+1
0093            TSAMP=BUF(N)
0094            N=N+1
0095            ICONT=BUF(N)
0096            N=N+1
0097            NEXPL=BUF(N)
0098            IF(NEXPL.EQ.0) GO TO 10
0099            DO 17 J=1,NEXPL
0100            N=N+1
0101            EXPL(J)=BUF(N)
0102    17      CONTINUE
0103            GO TO 10
        C
        C  CARD TYPE 3---SOILS
0104    30      IF(ITYPE.EQ.0) GO TO 35
0105            IPLACE=3
```

```
0106              GO TO(60,70,80,90),ITYPE
0107     35       ITYPE=3
0108              N=N+1
0109              TIM3=BUF(N)
0110              N=N+1
0111              DEPTH3=BUF(N)
0112              N=N+1
0113              NSOIL=BUF(N)
0114              IF(NSOIL.EQ.0) GO TO 10
0115              DO 36 J=1,NSOIL
0116              N=N+1
0117              SOILS(J)=BUF(N)
0118     36       CONTINUE
0119              GO TO 10
          C
          C  CARD TYPE 4---COMMENTS
0120     50     IF(ITYPE.EU.0) GO TO 55
          C  PLOT LAST CARD TYPE
0121              IPLACE=4
0122              GO TO(60,70,80,90),ITYPE
0123     55       ITYPE=4
0124              N=N+1
0125              TC(1)=BUF(N)
0126              N=N+1
0127              TC(2)=BUF(N)
0128              N=N+1
0129              CDEP=BUF(N)
0130              N=N+1
0131              NCOM=BUF(N)
0132              IF(NCOM.EQ.0) GO TO 10
0133              DO 56 J=1,NCOM
0134              N=N+1
0135              COMM(J)=BUF(N)
0136     56       CONTINUE
0137              GO TO 10
          C
          C
          C
          C *******  PLOT LAST DATA TYPE  *********
          C
          C
          C   CARD TYPE 1
          C
0138     60       NCARD1=NCARD1+1
0139              IF(NCARD1.NE.1) GO TO 63
0140              CALL DOTCHR(130,40,DDATE,6,SIZ,1)
0141              DO 61 J=1,6
0142              STRING(J)=DRL(J)
0143     61       CONTINUE
0144              CALL DOTCHR(140,40,STRING,24,SIZ,1)
0145              DO 62 J=1,6
0146              STRING(J)=DWN(J)
0147     62        CONTINUE
0148              CALL DOTCHR(170,40,STRING,18,SIZ,1)
0149     63       GO TO(13,16,35,55,9950),IPLACE
          C
```

```
        C
        C   CARD TYPE 2
        C
0150    70      IF(NEX.EQ.0) GO TO 74
0151            IF(NEX.NE.NEXPL) GO TO 72
0152            DO 71 J=1,NEX
0153            IF(ELAST(J).NE.EXPL(J)) GO TO 72
0154    71      CONTINUE
        C  SAME
0155            GO TO 77
        C  NEW STRATA--PLOT LAST ONE
0156    72      X=(DEPTHE+DEPTH)/(2.*DSCALE) + 200
0157            XT=(DEPTHE+.7)/DSCALE + 200
0158            XB=(DEPTH-.7)/DSCALE + 200
0159            CALL DOTLIN(XT,190,XT,160)
0160            CALL DOTLIN(XT,160,X,160)
0161            CALL DOTLIN(X,135,X,160)
0162            CALL DOTLIN(X,160,XB,160)
0163            CALL DOTLIN(XB,160,XB,190)
0164            LE=0
0165            DO 73 J=1,NEX
0166            IF(ELAST(J).EQ.BLANKS) GO TO 73
0167            LE=LE+1
0168            STRING(J)=ELAST(J)
0169    73      CONTINUE
0170            LEN=LE*4
0171            XX=X-5
0172            CALL DOTCHR(XX,8,STRING,LEN,SIZ,1)
0173    74      IF(NEXPL.EQ.0) GO TO 76
0174            DO 75 J=1,NEXPL
0175            ELAST(J)=EXPL(J)
0176    75      CONTINUE
0177    76      NEX=NEXPL
0178    77      DEPTHE=DEPTH
0179            GO TO(13,16,35,55,9950,9960),IPLACE
        C
        C
        C   CARD TYPE 3
        C
        C  CHECK FOR SOIL
0180    80      IF(NS.EQ.0) GO TO 84
0181            IF(NS.NE.NSOIL) GO TO 82
0182            DO 81 J=1,NSOIL
0183            IF(SLAST(J).NE.SOILS(J)) GO TO 82
0184    81      CONTINUE
        C   SAME SOIL
0185            TIME1=TIM2
0186            GO TO 87
        C  NEW SOIL--PLOT LAST ONE
0187    82      X=DEPTHL/DSCALE + 200
0188            CALL DOTLIN(X,200,X,264)
0189            ENCODE(6,682,A) DEPTHL
0190    682     FORMAT(F6.1)
0191            CALL DOTCHR(X,265,A,6,SIZ,1)
0192            X=(PREVD+DEPTHL)/(2.*DSCALE) + 200
0193            LS=0
```

```
0194              DO 83 J=1,NS
0195              IF(SLAST(J),EQ,BLANKS) GO TO 83
0196              LS=LS+1
0197              STRING(LS)=SLAST(J)
0198     83     CONTINUE
0199              LEN=LS*4
0200              CALL DOTCHR(X,310,STRING,LEN,SIZ,1)
0201              PREVD=DEPTHL
0202     84       IF(NSOIL,EQ,0) GO TO 86
0203              DO 85 J=1,NSOIL
0204              SLAST(J)=SOILS(J)
0205     85       CONTINUE
0206     86       NS=NSOIL
0207     87       DEPTHL=DEPTH3
0208              GO TO(13,16,35,55,9950,9990),IPLACE
         C
         C
         C  CARD TYPE 4
         C
         C    CHECK FOR COMMENT
0209     90       X=CDEP/DSCALE + 200
0210              LC=0
0211              IF(NCOM,EQ,0) GO TO 97
0212              DO 95 J=1,NCOM
0213              IF(COMM(J),EQ,BLANKS) GO TO 95
0214              LC=LC+1
0215              STRING(J)=COMM(J)
0216     95       CONTINUE
0217              LEN=LC*4
0218              NCH=0
0219              DO 96 J=1,LEN
0220              NCH=NCH+1
0221              STRNG2(NCH)=STRNG1(J)
0222              IF(NCH,LT,80) GO TO 96
0223              IF(STRNG2(NCH),NE,BLANK) GO TO 96
         C    WRITE THIS LINE
0224              CALL DOTCHR(X,600,STRNG2,NCH,SIZ,1)
0225              NCH=0
0226              X=X+10
0227     96       CONTINUE
0228              IF(NCH,GE,80) GO TO 97
0229              CALL DOTCHR(X,600,STRNG1,NCH,SIZ,1)
0230     97       GO TO(13,16,35,55,9950),IPLACE
         C
         C
         C  ERROR IN DATA LENGTH
0231     98       WRITE(5,598)
0232     598      FORMAT(///,' ERROR IN DATA LENGTH  "WLGCAL"',///)
0233              GO TO 9990
         C  READ THRU END OF FILE
0234     99       IPLACE=5
0235              GO TO(50,70,80,90),ITYPE
0236     9950     IPLACE=6
0237              NEXPL=0
0238              DEPTH=TOTDEP
0239              GO TO 70
```

WLGGLD.FTN /TR:BLOCKS/WR

```
0240     9960     NSOIL=0
0241              GO TO 80
0242     9990     CALL DOTOUT
0243              CALL DOTDUN
0244              IF (ICNTRL .NE. 1) GO TO 9991
0245              IEFN = 33
0246              CALL SETEF (IEFN,IDS)
0247     9991     CONTINUE
0248              CALL EXIT
0249              END
```

PROGRAM SECTIONS

NAME	SIZE		ATTRIBUTES
$CODE1	004630	1228	RW,I,CON,LCL
$PDATA	000412	133	RW,D,CON,LCL
$IDATA	000716	231	RW,D,CON,LCL
$VARS	030762	6393	RW,D,CON,LCL
$TEMPS	000014	6	RW,D,CON,LCL
HDR	000776	255	RW,D,OVR,GBL
BLK2	013560	3000	RW,D,OVR,GBL

TOTAL SPACE ALLOCATED = 053734 11246

```
,LP=DP1:WLGGLD
```

WLGGLD2.FTN

DRILLERS LOG (GOULD-SHADED)

```
C ***************  DB0:[351,100]WLGGLD2,FTN  ***************
C
C
C  PROGRAM ID:  SRV-11-WEL-3-4  (TASK IMAGE IS NAMED WLGTGP)
C
C  PROGRAM DESCRIPTION:
C       THIS PROGRAM OUTPUTS DRILLERS LOGS TO GOULD PLOTTER,
C       DATA OUTPUT: STRATIFICATION, SOIL TYPES, AND COMMENTS,
C       WELL SCALE = 20 FEET PER INCH,  THIS VERSION OUTPUTS
C       SOIL TYPE SYMBOLS (SHADING),
C              "HANFORD RESERVATION"
C
C  DATA FILES:
C       NAME                    LUN             TYPE            ACCESS
C       FILE U--FRMTWLG         1               RAN             R
C       GOULD OUTPUT            7                               W
C
C
C       MCR>TKB @DB0:[351,100]WLGGLD2
C
C
C  BATTELLE MEMORIAL INSTITUTE
C  PACIFIC NORTHWEST LABORATORIES
C  WATER & LAND RESOURCES DEPT.
C
C  AUTHOR(S):    DR FRIEDRICHS
C
C
C  DATE:         INITIAL VERSION   JANUARY 1977
C                CURRENT VERSION   APRIL   1978
C
C
C
0001            BYTE WLDES
0002            BYTE DAT(9),TIM(8),STRNG1(720),STRNG2(100),BLANK,D(48)
0003            BYTE A(6),ASOIL(4),SHADE1,SHADE2
C
0004            INTEGER BUFFER,X,Y,XX,YY,SIZ,BOT,XT,XB
C
0005            DIMENSION SOILS(180),STRING(180),SLAST(180)
0006            DIMENSION EXPL(100),ELAST(100),COMM(200),INODE(253)
0007            DIMENSION NDUM(2000),NBUF(256),IBUF(255)
0008            DIMENSION DDATE(2),T(2),TC(2)
C
0009            COMMON/HDR/ IDM(2),WLDES(12),IDUM1(10),CASE,IDUM2(21),IBLK,
                1           NSTRT,NWRD,IDUM3(187),OWN(6),DRL(6)
0010            COMMON/BLK2/ BUF(1000)
0011            COMMON/BFR/  BUFFER(12288)
C
0012            EQUIVALENCE (BUF(1),NDUM(1)),(TIM3,ASOIL(1))
0013            EQUIVALENCE (STRING(1),STRNG1(1))
0014            EQUIVALENCE (INODE(1),WLDES(1)),(INODE(1),IBUF(3))
C
0015            DATA BLANKS /4H    /
0016            DATA BLANK  /1H /
C
```

```
        C
        C
        C  READ TRANSFER NODE
        C       TASK = RAD50('MNTR45')
0017            CALL VRECSP (TASK,IBUF,255,,,IOS)
0018            JPNT = 0
0019            NDEL = 0
        C
0020            CALL ASNLUN(1,'DB',0)
0021            CALL OPFILE(1,'FRMTWLG',8000,,IPNT1)
        C
0022            NWORD=NWRD
0023            KBLK=IBLK
0024            JB=NSTRT
0025            STRT=KBLK
0026            CALL OPR(IPNT1,STRT,NBUF,256,)
0027            DO 6 J=1,NWORD
0028            JB=JB+1
0029            NDUM(J)=NBUF(JB)
0030            IF(JB.LT.256) GO TO 6
0031            KBLK=KBLK+1
0032            STRT=KBLK
0033            CALL OPR(IPNT1,STRT,NBUF,256,)
0034            JB=0
0035    6       CONTINUE
        C
0036            NTOT=BUF(4)
0037            DDATE(1)=BUF(5)
0038            DDATE(2)=BUF(6)
0039            TOTDEP=BUF(7)
        C  SET SCALE AT 20FT/IN
0040            DSCALE=20./100.
0041            BOT=TOTDEP/DSCALE + 200
0042            NN=TOTDEP/20.
0043            NN=NN*100 + 200
0044            NN=(NN/32+2)*32
        C
        C
0045            CALL TIME(TIM)
0046            CALL DATE(DAT)
0047            CALL DOTINI(7,BUFFER,96,1024)
0048            DO 9990 INDEX=1,NN,96
0049            X=100
0050            Y=100
0051            SIZ=24
0052            CALL DOTCHR(X,Y,'WELL NO.',,SIZ,1)
        C   PLOT WELL DESIGNATION
0053            CALL DOTCHR(X,230,WLDES,12,SIZ,1)
0054            X=190
0055            SIZ=16
0056            CALL DOTCHR(X,8,'STRATIGRAPHY',,SIZ,1)
0057            CALL DOTCHR(X,300,'MATERIALS PENETRATED',,SIZ,1)
0058            CALL DOTCHR(X,660,'COMMENTS',,SIZ,1)
0059            X=100
0060            SIZ=8
0061            CALL DOTCHR(X,520,DAT,9,SIZ,1)
```

```
0062              CALL DOTCHR(X,610,TIM,0,SIZ,1)
0063              X=200
0064              CALL DOTLIN(X,750,X,260)
0065              CALL DOTLIN(X,188,X,0)
          C
0066              N=7
0067              CALL DOTLIN(BOT,188,170,188)
0068              CALL DOTLIN(170,188,170,192)
0069              CALL DOTLIN(170,192,BOT,192)
0070              CALL DOTLIN(BOT,256,170,256)
0071              CALL DOTLIN(170,256,170,260)
0072              CALL DOTLIN(170,260,BOT,260)
          C
          C
0073              PREVD=0.0
0074              NS=0
0075              NEX=0
0076              SHADE1='X'
0077              DPBOT=0.0
0078              ITYPE=0
0079              NCARD1=0
0080              SIZ=8
          C
          C
          C  ****** READ DATA FROM BUFFER ******
          C
0081      10      N=N+1
0082              IF(N.GT.NTOT) GO TO 98
0083              I1=BUF(N)
0084              GO TO(12,15,30,50,99),I1
0085      12      IF(ITYPE.EQ.0) GO TO 13
0086              IPLACE=1
0087              GO TO(60,70,80,90),ITYPE
          C
          C  CARD TYPE 1---HEADER INFO
0088      13      ITYPE=1
0089               GO TO 10
          C  CARD TYPE 2---STRATIFICATION
0090      15      IF(ITYPE.EQ.0) GO TO 16
0091              IPLACE=2
0092              GO TO (60,70,80,90),ITYPE
0093      16      ITYPE=2
0094              N=N+1
0095              TIM2=BUF(N)
0096              N=N+1
0097              DEPTH=BUF(N)
0098              N=N+1
0099              ISAMP=BUF(N)
0100              N=N+1
0101              ICONT=BUF(N)
0102              N=N+1
0103              NEXPL=BUF(N)
0104              IF(NEXPL.EQ.0) GO TO 10
0105              DO 17 J=1,NEXPL
0106              N=N+1
0107              EXPL(J)=BUF(N)
```

```
0108     17       CONTINUE
0109              GO TO 10
         C
         C  CARD TYPE 3---SOILS
0110     30       IF(ITYPE.EQ.0) GO TO 35
0111              IPLACE=3
0112              GO TO(60,70,80,90),ITYPE
0113     35       ITYPE=3
0114              N=N+1
0115              TIM3=BUF(N)
0116              SHADE2=ASOIL(1)
0117              N=N+1
0118              DEPTH3=BUF(N)
0119              N=N+1
0120              NSOIL=BUF(N)
0121              IF(NSOIL.EQ.0) GO TO 10
0122              DO 36 J=1,NSOIL
0123              N=N+1
0124              SOILS(J)=BUF(N)
0125     36       CONTINUE
0126              GO TO 10
         C
         C  CARD TYPE 4---COMMENTS
0127     50     IF(ITYPE.EQ.0) GO TO 55
         C  PLOT LAST CARD TYPE
0128              IPLACE=4
0129              GO TO(60,70,80,90),ITYPE
0130     55       ITYPE=4
0131              N=N+1
0132              TC(1)=BUF(N)
0133              N=N+1
0134              TC(2)=BUF(N)
0135              N=N+1
0136              CDEP=BUF(N)
0137              N=N+1
0138              NCOM=BUF(N)
0139              IF(NCOM.EQ.0) GO TO 10
0140              DO 56 J=1,NCOM
0141              N=N+1
0142              COMM(J)=BUF(N)
0143     56       CONTINUE
0144              GO TO 10
         C
         C
         C
         C *******  PLOT LAST DATA TYPE  *********
         C
         C
         C   CARD TYPE 1
         C
0145     60       NCARD1=NCARD1+1
0146              IF(NCARD1.NE.1) GO TO 63
0147              CALL DOTCHR(130,40,DDATE,6,SIZ,1)
0148              DO 61 J=1,6
0149              STRING(J)=DRL(J)
0150     61       CONTINUE
```

```
0151            CALL DOTCHR(140,40,STRING,24,SIZ,1)
0152            DO 62 J=1,6
0153            STRING(J)=OWN(J)
0154    62       CONTINUE
0155            CALL DOTCHR(170,40,STRING,18,SIZ,1)
0156    63      GO TO(13,16,35,55,9950),IPLACE
        C
        C
        C   CARD TYPE 2
        C
0157    70      IF(NEX.EQ.0) GO TO 74
0158            IF(NEX.NE.NEXPL) GO TO 72
0159            DO 71 J=1,NEX
0160            IF(ELAST(J).NE.EXPL(J)) GO TO 72
0161    71      CONTINUE
        C  SAME
0162            GO TO 77
        C  NEW STRATA--PLOT LAST ONE
0163    72      X=(DEPTHE+DEPTH)/(2.*DSCALE) + 200
0164            XT=(DEPTHE+.7)/DSCALE + 200
0165            XB=(DEPTH-.7)/DSCALE + 200
0166            CALL DOTLIN(XT,178,XT,160)
0167            CALL DOTLIN(XT,160,X,160)
0168            CALL DOTLIN(X,135,X,160)
0169            CALL DOTLIN(X,160,XB,160)
0170            CALL DOTLIN(XB,160,XB,178)
0171            LE=0
0172            DO 73 J=1,NEX
0173            IF(ELAST(J).EQ.BLANKS) GO TO 73
0174            LE=LE+1
0175            STRING(J)=ELAST(J)
0176    73      CONTINUE
0177            LEN=LE*4
0178            XX=X-5
0179            CALL DOTCHR(XX,8,STRING,LEN,SIZ,1)
0180    74      IF(NEXPL.EQ.0) GO TO 76
0181            DO 75 J=1,NEXPL
0182            ELAST(J)=EXPL(J)
0183    75      CONTINUE
0184    76      NEX=NEXPL
0185    77      DEPTHE=DEPTH
0186            GO TO(13,16,35,55,9950,9960),IPLACE
        C
        C
        C   CARD TYPE 3
        C
        C   CHECK SHADING MATERIAL
0187    80      IF(SHADE1.EQ.SHADE2) GO TO 8050
0188            IF(SHADE1.EQ.'G') CALL GRAVEL(DPTOP,DPBOT,DSCALE)
0189            IF(SHADE1.EQ.'B') CALL BASALT(DPTOP,DPBOT,DSCALE)
0190            IF(SHADE1.EQ.'S') CALL SAND(DPTOP,DPBOT,DSCALE)
0191            IF(SHADE1.EQ.'M') CALL MUD(DPTOP,DPBOT,DSCALE)
0192    8010    DPTOP=DPBOT
0193            SHADE1=SHADE2
0194    8050    DPBOT=DEPTH3
        C
```

```
        C  CHECK FOR SOIL
0195            IF(NS.EQ.0) GO TO 84
0196            IF(NS.NE.NSOIL) GO TO 82
0197            DO 81 J=1,NSOIL
0198            IF(SLAST(J).NE.SOILS(J)) GO TO 82
0199    81      CONTINUE
        C   SAME SOIL
0200            TIMEL=TIM2
0201            GO TO 87
        C  NEW SOIL--PLOT LAST ONE
0202    82      X=DEPTHL/DSCALE + 200
0203            CALL DOTLIN(X,180,X,260)
0204            ENCODE(6,682,A) DEPTHL
0205    682     FORMAT(F6.1)
0206            CALL DOTCHR(X,261,A,6,SIZ,1)
0207            X=(PREVD+DEPTHL)/(2.*DSCALE) + 200
0208            LS=0
0209            DO 83 J=1,NS
0210            IF(SLAST(J).EQ.BLANKS) GO TO 83
0211            LS=LS+1
0212            STRING(LS)=SLAST(J)
0213    83    CONTINUE
0214            LEN=LS*4
0215            CALL DOTCHR(X,306,STRING,LEN,SIZ,1)
0216            PREVD=DEPTHL
0217    84      IF(NSOIL.EQ.0) GO TO 86
0218            DO 85 J=1,NSOIL
0219            SLAST(J)=SOILS(J)
0220    85      CONTINUE
0221    86      NS=NSOIL
0222    87      DEPTHL=DEPTH3
0223            GO TO(13,16,35,55,9950,9990),IPLACE
        C
        C
        C  CARD TYPE 4
        C
        C   CHECK FOR COMMENT
0224    90      X=CDEP/DSCALE + 200
0225            LC=0
0226            IF(NCOM.EQ.0) GO TO 97
0227            DO 95 J=1,NCOM
0228            IF(COMM(J).EQ.BLANKS) GO TO 95
0229            LC=LC+1
0230            STRING(J)=COMM(J)
0231    95      CONTINUE
0232            LEN=LC*4
0233            NCH=0
0234            DO 96 J=1,LEN
0235            NCH=NCH+1
0236            STRNG2(NCH)=STRNG1(J)
0237            IF(NCH.LT.80) GO TO 96
0238            IF(STRNG2(NCH).NE.BLANK) GO TO 96
        C   WRITE THIS LINE
0239            CALL DOTCHR(X,600,STRNG2,NCH,SIZ,1)
0240            NCH=0
0241            X=X+10
```

```
FORTRAN IV-PLUS V02-51D          14125107     11-SEP-79           PAGE 7
WLGGLD2.FTN      /TR:BLOCKS/WR

0242     96      CONTINUE
0243             IF(NCH.GE.80) GO TO 97
0244             CALL DOTCHR(X,600,STRNG1,NCH,SIZ,1)
0245     97      GO TO(13,16,35,55,9950),IPLACE
         C
         C
         C  ERROR IN DATA LENGTH
0246     98      WRITE(5,598)
0247     598     FORMAT(///,' ERROR IN DATA LENGTH  "WLGCAL"',///)
0248             GO TO 9990
         C  READ THRU END OF FILE
0249     99      IPLACE=5
0250             GO TO(60,70,80,90),ITYPE
0251     9950    IPLACE=6
0252             NEXPL=0
0253             DEPTH=TOTDEP
0254             GO TO 70
0255     9960    NSOIL=0
0256             SHADE2='X'
0257             GO TO 80
0258     9990    CALL DOTOUT
0259             CALL DOTDUN
0260             CALL EXIT
0261             END


PROGRAM SECTIONS

 NAME        SIZE                  ATTRIBUTES

$CODE1   005000  1280              RW,I,CON,LCL
$PDATA   000416   135              RW,D,CON,LCL
$IDATA   000720   232              RW,D,CON,LCL
$VARS    010776  2303              RW,D,CON,LCL
$TEMPS   000014     6              RW,D,CON,LCL
HDR      000776   255              RW,D,OVR,GBL
BLK2     007640  2000              RW,D,OVR,GBL
BFR      060000 12288              RW,D,OVR,GBL


TOTAL SPACE ALLOCATED = 110206 18499
```

```
FORTRAN IV-PLUS V02-51D          14126107     11-SEP-79          PAGE 8
WLGGLD2.FTN      /TR:BLOCKS/WR

        C
        C
0001                SUBROUTINE GRAVEL(DPTOP,DPBOT,DSCALE)
        C
0002                INTEGER BUFFER
0003                COMMON/BFR/ BUFFER(12288)
        C
0004                IX=DPTOP/DSCALE + 240
0005                NFEET=DPBOT-DPTOP
0006                NTOP=((NFEET+1)/2)*2
0007                DO 30 J=2,NTOP,2
0008                IY=192
0009                DO 20 I=1,8
0010                CALL DOTLIN(IX+2,IY+3,IX+2,IY+6)
0011                CALL DOTLIN(IX+2,IY+6,IX+4,IY+6)
0012                CALL DOTLIN(IX+4,IY+6,IX+8,IY+3)
0013                CALL DOTLIN(IX+4,IY+3,IX+2,IY+3)
0014                IF(J.GT.NFEET) GO TO 19
0015                CALL DOTLIN(IX+6,IY,IX+6,IY+3)
0016                CALL DOTLIN(IX+6,IY+3,IX+10,IY+3)
0017                CALL DOTLIN(IX+10,IY+3,IX+10,IY)
        C
0018                CALL DOTLIN(IX+6,IY+8,IX+6,IY+6)
0019                CALL DOTLIN(IX+6,IY+6,IX+10,IY+6)
0020                CALL DOTLIN(IX+10,IY+6,IX+10,IY+8)
0021    19          IY=IY+8
0022    20          CONTINUE
0023                IX=IX+10
0024    30          CONTINUE
0025                RETURN
0026                END


PROGRAM SECTIONS

 NAME        SIZE                    ATTRIBUTES

$CODE1   001310    356              RW,I,CON,LCL
$IDATA   000036     15              RW,D,CON,LCL
$VARS    000014      6              RW,D,CON,LCL
$TEMPS   000012      5              RW,D,CON,LCL
BFR      060000  12288              RW,D,OVR,GBL


TOTAL SPACE ALLOCATED = 061374 12670
```

```
FORTRAN IV-PLUS V02-51D          14:27:00     11-SEP-79          PAGE 9
WLGGLD2.FTN      /TR:BLOCKS/WR

        C
        C
0001                SUBROUTINE SAND(DPTOP,DPBOT,DSCALE)
        C
0002                INTEGER BUFFER
0003                COMMON/BFR/ BUFFER(12288)
        C
0004                IX=DPTOP/DSCALE + 200
0005                NFEET=DPBOT-DPTOP
0006                NTOP=((NFEET+1)/2)*2
0007                DO 30 J=2,NTOP,2
0008                IY=192
0009                DO 20 I=1,8
0010                CALL DOTLIN(IX+1,IY+2,IX+1,IY+3)
0011                CALL DOTLIN(IX+1,IY+6,IX+1,IY+7)
0012                CALL DOTLIN(IX+2,IY+6,IX+2,IY+7)
0013                CALL DOTLIN(IX+3,IY+1,IX+3,IY+2)
0014                CALL DOTLIN(IX+4,IY+1,IX+4,IY+2)
0015                CALL DOTLIN(IX+4,IY+5,IX+4,IY+6)
0016                IF(J.GT.NFEET) GO TO 19
0017                CALL DOTLIN(IX+6,IY+7,IX+6,IY+8)
0018                CALL DOTLIN(IX+7,IY+7,IX+7,IY+8)
0019                CALL DOTLIN(IX+7,IY+3,IX+7,IY+4)
0020                CALL DOTLIN(IX+9,IY+5,IX+9,IY+6)
0021                CALL DOTLIN(IX+10,IY+2,IX+10,IY+3)
0022    19          IY=IY+8
0023    20          CONTINUE
0024                IX=IX+10
0025    30          CONTINUE
0026                RETURN
0027                END


PROGRAM SECTIONS

 NAME        SIZE                  ATTRIBUTES

$CODE1   001454    406              RW,I,CON,LCL
$IDATA   000012      5              RW,D,CON,LCL
$VARS    000014      6              RW,D,CON,LCL
$TEMPS   000016      7              RW,D,CON,LCL
BFR      060000 12288              RW,D,OVR,GBL


TOTAL SPACE ALLOCATED = 061520 12712
```

```
FORTRAN IV-PLUS V02-510          14:27:16     11-SEP-79          PAGE 10
WLGGL02.FTN      /TRIALOCKS/WR

        C
        C
0001            SUBROUTINE BASALT(DPTOP,DPBOT,DSCALE)
        C
0002            INTEGER BUFFER
0003            COMMON/BFR/ BUFFER(12288)
        C
0004            IX1=DPTOP/DSCALE + 200
0005            IX2=DPBOT/DSCALE + 200
0006            IY=195
0007            DO 30 J=1,8
0008            CALL DOTLIN(IX1,IY,IX2,IY)
0009            CALL DOTLIN(IX1,IY+1,IX2,IY+1)
0010            CALL DOTLIN(IX1,IY+2,IX2,IY+2)
0011            CALL DOTLIN(IX1,IY+3,IX2,IY+3)
0012            IY=IY+8
0013    30      CONTINUE
0014            RETURN
0015            END


PROGRAM SECTIONS

 NAME        SIZE                    ATTRIBUTES

$CODE1  000314    102           RW,I,CON,LCL
$IDATA  000024     10           RW,D,CON,LCL
$VARS   000010      4           RW,D,CON,LCL
BFR     060000  12288           RW,D,OVR,GBL


TOTAL SPACE ALLOCATED = 060350 12404
```

```
        C
        C
0001            SUBROUTINE MUD(DPTOP,DPBOT,DSCALE)
        C
0002            INTEGER BUFFER
0003            COMMON/BFR/ BUFFER(12288)
        C
0004            IX=DPTOP/DSCALE + 200
0005            NFEET=DPBOT-DPTOP
0006            NTOP=((NFEET+1)/2)*2
0007            DO 30 J=2,NTOP,2
0008            IY=192
0009            DO 20 I=1,8
0010            CALL DOTLIN(IX+3,IY+3,IX+3,IY+8)
0011            IF(J.GT.NFEET) GO TO 19
0012            CALL DOTLIN(IX+6,IY+1,IX+6,IY+6)
0013            CALL DOTLIN(IX+9,IY+1,IX+9,IY+4)
0014            CALL DOTLIN(IX+9,IY+7,IX+9,IY+8)
0015    19      IY=IY+8
0016    20      CONTINUE
0017            IX=IX+10
0018    30      CONTINUE
0019            RETURN
0020            END
```

PROGRAM SECTIONS

NAME	SIZE		ATTRIBUTES
$CODE1	000634	206	RW,I,CON,LCL
$IDATA	000012	5	RW,D,CON,LCL
$VARS	000014	6	RW,D,CON,LCL
$TEMPS	000006	3	RW,D,CON,LCL
BFR	060000	12288	RW,D,OVR,GBL

TOTAL SPACE ALLOCATED = 060670 12508

```
,LP=DP1:WLGGLD2
```

TMPCAL.FLX

TEMPERATURE (CALCOMP)

```
(FLECS VERSION 22.46)     11-SEP-79   14:29:04  PAGE  0001

      ----------------------------------------

00001   C ***************  [351,100]TMPCAL.FLX  ***************
00002   C
00003   C
00004   C  PROGRAM ID: SRV-11-WEL-10-1 ARREVIATED WELL DESIG. VERSION
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL TEMPERATURE DATA FROM
00008   C     THE CIRMIS DATA BANK AND DRAW A CAL-COMP PLOT OF THE DATA.  THIS
00009   C     PROGRAM IS STARTED BY 'RNTR13' AND RECEIVES CONTROL DATA VIA THE
00010   C     TRANSFER NODE (MEMORY TO MEMORY).
00011   C
00012   C  DATA FILES:
00013   C     NAME                      LUN             TYPE            ACCESS
00014   C     FILE O---F RMTTMP          1              RAN             R
00015   C     MM-------TNODE             2              MM              R
00016   C
00017   C
00018   C  LOADING SEQUENCE:
00019   C
00020   C     MCR>TKB @CMD:[351,100]TMPCAL.CMD
00021   C
00022   C
00023   C  BATTELLE MEMORIAL INSTITUTE
00024   C  PACIFIC NORTHWEST LABORATORIES
00025   C  WATER & LAND RESOURCES DEPT.
00026   C
00027   C  AUTHOR(S):    DR FRIEDRICHS
00028   C                DW DAMSCHEN
00029   C
00030   C  DATE:         INITIAL VERSION DECEMBER 1977
00031     C              CURRENT VERSION  AUGUST  1979
00032   C
00033   C
00034   C
00035         BYTE STAT,WLDES
00036   C
00037         DIMENSION H(512),T(512),INODE(255),ITRAN(255)
00038   C
00039         COMMON/HDR/ IDM(2),WLDES(12),IDM2(249)
00040   C
00041         EQUIVALENCE (WLDES(1),INODE(1)),(CASELV,INODE(17))
00042         EQUIVALENCE (IDKV,INODE(55)),(NPTV,INODE(56)),(IDTYP,INODE(255))
00043         EQUIVALENCE (IDM(1),ITRAN(1))
00044   C
00045   C
00046   C *** READ THE TRANSFER NODE ***
00047   C
00048         CALL VRECSP (TASK,ITRAN,255,,,IDS)
00049   C
00050         IUNIT = 4
00051   C
00052         CALL ASNLUN(1,'DR',IUNIT)
00053         CALL OPFILE(1,'FRMTTMP',4000,,IPNT)
```

```
(FLECS VERSION 22.46)     11-SEP-79   14129104  PAGE  00002

00054   C
00055   C *** RETRIEVE THE TEMPERATURE DATA ***
00056   C
00057         RETRIEVE-THE-TEMPERATURE-DATA
00058   C
00059   11    IF(INODE(35).NE.0) GO TO 13
00060   C
00061   C *** CALCULATE AUTO SCALE FACTORS ***
00062   C
00063         TMAX=0.0
00064         TMIN=999999.
00065         HMAX=0.0
00066         HMIN=1000.
00067         DO(I=1,NPTS)
00068         .  IF(T(I).GT.TMAX) TMAX=T(I)
00069         .  IF(T(I).LT.TMIN) TMIN=T(I)
00070         .  IF(H(I).GT.HMAX) HMAX=H(I)
00071         .  IF(H(I).LT.HMIN) HMIN=H(I)
00072         ...FIN
00073         MINX=IFIX(TMIN/10000.)
00074         MAXX=IFIX(TMAX/10000.)
00075         MINY=IFIX(HMIN/10.)*10-10
00076         MAXY=IFIX(HMAX/10.)*10+10
00077         INCY=10
00078         IF((MAXY-MINY).LE.100) INCY=5
00079         IF((MAXY-MINY).LE.40) INCY=2
00080         GO TO 14
00081   13    MINX=INODE(36)
00082         MINY=INODE(37)
00083         MAXX=INODE(38)
00084         MAXY=INODE(39)
00085         INCY=INODE(40)
00086   14    IMN=90
00087         IMX=890
00088         JMN=90
00089         JMX=890
00090         SCLX=IMX-IMN
00091         SCLY=JMX-JMN
00092   C
00093   C *** DRAW THE PLOT ***
00094   C
00095         CALL PLOTS(0.,0.,7)
00096         CALL NEWPEN(1)
00097         CALL FEDCL(0.0,0.0,1)
00098         CALL PLOT(1.0,1.5,-3)
00099         CALL FACTOR(.75)
00100         SIZ=0.2
00101         SIZ2=0.09
00102   C---DEFINE BACKGROUND AND PLOT IT
00103         CALL NEWPEN(1)
00104         X1=FLOAT(IMN)/100.
00105         X2=FLOAT(IMX)/100.
00106         Y1=FLOAT(JMN)/100.
00107         Y2=FLOAT(JMX)/100.
00108         CALL PLOT(X1,Y1,+3)
00109         CALL PLOT(X1,Y2,2)
```

```
 00110          CALL PLOT(X2,Y2,2)
 00111          CALL PLOT(X2,Y1,2)
 00112          CALL PLOT(X1,Y1,2)
 00113          CALL NEWPEN(1)
 00114          CALL SYMBOL(3.7,.25,SIZ,'CALENDAR YEAR',0.,13)
 00115          CALL SYMBOL(.23,2.85,SIZ,'TEMPERATURE (DEGREES C)',90.,23)
 00116          CALL SYMBOL(3.3,9.6,SIZ,'TEMPERATURE HISTORY',0.,19)
 00117          CALL NEWPEN(1)
 00118          CALL SYMBOL(3.8,9.4,SIZ2,'WELL DESIGNATION = ',0,0,19)
 00119          CALL SYMBOL(5.3,9.4,SIZ2,WLDES,0.0,12)
 00120          CALL SYMBOL(3.8,9.2,SIZ2,'CASING ELEVATION =',0,0,18)
 00121          CALL NUMBER(5.3,9.2,SIZ2,CASELV,0.0,2)
 00122          NDIF=MAXY-MINY
 00123          YSCALE=SCLY/FLOAT(NDIF)
 00124    C----DRAW Y LINES FOR CHART
 00125          CALL NEWPEN(1)
 00126          NN=MINY
 00127          JDD=1
 00128          NDIF=NDIF/INCY+1
 00129          DO(JD=1,NDIF)
 00130           . Y=(FLOAT(NN-MINY)*YSCALE+FLOAT(JMN))/100.
 00131           . IF(JDD.NE.1) GO TO 3602
 00132           . IF(JD.EQ.1.OR.JD.EQ.NDIF) GO TO 3601
 00133           . CALL PLOT(X1,Y,3)
 00134           . CALL PLOT(X2,Y,2)
 00135    3601   . YV=FLOAT(NN)
 00136           . CALL NUMBER(X1-.3,Y,SIZ2,YV,0.,-1)
 00137           . GO TO 3603
 00138    3602   . CALL PLOT(X1,Y,3)
 00139           . CALL PLOT(X1+.1,Y,2)
 00140    3603   . JDD=JDD+1
 00141           . IF(JDD.EQ.6) JDD=1
 00142           . NN=NN+INCY
 00143           ...FIN
 00144    C----DRAW X LINES FOR CHART
 00145    3610   NDIF=(MAXX-MINX) + 1
 00146          XSCALE=SCLX/(FLOAT(NDIF)*365.)
 00147          XINC=(SCLX/FLOAT(NDIF))/100.
 00148          XADJ=XINC/2. + SIZ2*.86
 00149          JDD=1
 00150          DO(JD=MINX,MAXX)
 00151           . X=FLOAT(JD-MINX+1)*XINC + X1
 00152           . XV=JD
 00153           . CALL NUMBER(X-XADJ,Y1-.12,SIZ2,XV,0.0,-1)
 00154           . IF(JD.EQ.MAXX) GO TO 3650
 00155           . GO TO(3620,3630),JDD
 00156    3620   . JDD=2
 00157           . CALL PLOT(X,Y1,3)
 00158           . CALL PLOT(X,Y1+.1,2)
 00159           . GO TO 3650
 00160    3630   . JDD=1
 00161           . CALL PLOT(X,Y1,3)
 00162           . CALL PLOT(X,Y2,2)
 00163    3650   . CONTINUE
 00164          ...FIN
 00165    C---DRAW THE TEMPERATURE CURVE
```

```
00166          NABSV=0
00167          CALL NEWPEN(1)
00168          NABSV=0
00169          DO(JK=1,NPTS)
00170           . DATE=T(JK)
00171           . ELV=H(JK)
00172           . Y=ELV
00173           . IYR=DATE/10000.
00174           . MO=DATE/100.-FLOAT(IYR)*100.
00175           . IDA=DATE-FLOAT(MO)*100.-FLOAT(IYR)*10000.
00176    38     . DAYS=(FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
00177           . X=(DAYS*XSCALE+FLOAT(IMN))/100.
00178           . Y=((Y-FLOAT(MINY))*YSCALE+FLOAT(JMN))/100.
00179    C      . CHECK FOR POINT WITHIN REGION
00180           . IF(X.LT.X1.OR.X.GT.X2) GO TO 3850
00181           . IF(Y.GE.Y1.AND.Y.LE.Y2) GO TO 39
00182    3850   . NABSV=0
00183           . GO TO 40
00184    C    IF LAST VALUE OUTSIDE REGION DRAW BLANK VECTOR
00185    39     . IF (NPTS.EQ.1) GO TO 3950
00186           . IF(NABSV.EQ.0) CALL PLOT(X,Y,3)
00187           . IF(NABSV.EQ.1) CALL PLOT(X,Y,2)
00188           . NABSV=1
00189           . GO TO 40
00190    3950   . CALL SYMBOL (X,Y,0.18,14,0.,-1)
00191    40     . CONTINUE
00192          ...FIN
00193    C
00194    C
00195    41     CALL FACTOR(1.0)
00196          CALL NEWPEN(1)
00197          CALL PLOT(-1.0,9.5,-3)
00198    45    CALL PLOTND
00199          CALL EXIT
00200    C

       ---------------------------------------

00201          TO RETRIEVE-THE-TEMPERATURE-DATA
00202    C       .
00203    C    IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
00204    C    NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
00205    C    IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
00206    C    NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
00207    C    INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
00208    C       .
00209           . IDKIN=INODE(50)
00210           . NPTSIN=INODE(51)
00211           . IDKEX=INODE(52)
00212           . NPTSEX=INODE(53)
00213           . INALOC=INODE(54)
00214    C       .
00215    C    READ THE DATA IN THE INITIAL ALLOCATION
00216    C       .
00217           . HADSI=IDKIN
00218           . WRDI=NPTSIN*2.
```

```
(FLECS VERSION 22.46)     11-SEP-79   14:29:04  PAGE  00005

  00219           .  TADSI=HADSI+INALOC
  00220           .  CALL DPR(IPNT1,HADSI,H,WRDI)
  00221           .  CALL DPR(IPNT1,TADSI,T,WRDI)
  00222    C      .
  00223    C    READ THE EXTENDED ALLOCATION (IF ANY)
  00224    C      .
  00225           .  IF (NPTSEX .NE. 0)
  00226           .  .  HADSE=IOKEX
  00227           .  .  WRDE=NPTSEX*2.
  00228    C      .  .
  00229    C    FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
  00230    C    TEMPERATURE AND TIME
  00231    C      .  .
  00232           .  .  TADSE=HADSE+2.
  00233           .  .  NOX=NPTSIN+1
  00234           .  .  CALL DPR(IPNT1,HADSE,H(NOX),WRDE)
  00235           .  .  CALL DPR(IPNT1,TADSE,T(NOX),WRDE)
  00236           .  ...FIN
  00237    C      .
  00238           .  NPTS=NPTSIN + NPTSEX
  00239           ...FIN
  00240           END

        ----------------------------------------

              PROCEDURE CROSS-REFERENCE TABLE

        00201 RETRIEVE-THE-TEMPERATURE-DATA
              00057


        (FLECS VERSION 22.46)

        ----------------------------------------
```

TMPFPR.FLX

TEMPERATURE (LINE PRINTER)

```
00001    C ***************  DB0:[351,100]TMPLPR.FLX  ***************
00002    C
00003    C
00004    C  PROGRAM ID: SRV-11-WEL-10-2  ABREVIATED WELL DESIG. VERSION
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C       THIS PROGRAM IS DESIGNED TO RETRIEVE WELL TEMPERATURE DATA FROM T
00008    C       CIRMIS DATA BANK AND PRINT THE DATA IN FORMATED FORM ON THE LINE
00009    C       PRINTER.  THIS PROGRAM IS STARTED BY 'MNTR13' AND RECEIVES CONTRO
00010    C       DATA VIA THE TRANSFER NODE DP1:[22,33]TNODE.RAN.
00011    C
00012    C  DATA FILES:
00013    C       NAME                    LUN             TYPE            ACCESS
00014    C       FILE O--FRMTTMP          1              RAN               R
00015    C       MM------TNODE            2              MM                R
00016    C
00017    C
00018    C  LOADING SEQUENCE:
00019    C
00020    C       MCR>TKB @DB0:[351,100]TMPLPR
00021    C
00022    C
00023    C  BATTELLE MEMORIAL INSTITUTE
00024    C  PACIFIC NORTHWEST LABORATORIES
00025    C  WATER & LAND RESOURCES DEPT.
00026    C
00027    C  AUTHOR(S):    DR FRIEDRICHS
00028    C                 DW DAMSCHEN
00029    C
00030    C             MODIFICATIONS FOR TERMINAL OUTPUT BY RC ARNETT, ARHCO
00031    C
00032    C
00033    C  DATE:          INITIAL VERSION  DECEMBER  1977
00034    C                 CURRENT VERSION   AUGUST   1979
00035    C
00036    C
00037    C
00038          BYTE DT(512,9),AMON(3,12),YR(2),DAY(2),STAT,WNAME(12)
00039          DIMENSION M(512),T(512),INODE(258)
00040    C
00041          EQUIVALENCE (WNAME(1),INODE(1))
00042          EQUIVALENCE (CASELV,INODE(17)),(IUNIT,INODE(258))
00043    C
00044          DATA AMON/'J','A','N','F','E','B','M','A','R','A','P','R',
00045         1 'M','A','Y','J','U','N','J','U','L','A','U','G','S','E','P',
00046         2 'O','C','T','N','O','V','D','E','C'/
00047    C
00048    C
00049    C *** READ THE TRANSFER NODE ***
00050    C
00051    C       TASK  =  RAD50('MNTR45')
00052          READ(2) TASK,INODE
00053    C       INODE(200) IS A FLAG, 5 = TI: TERMINAL, ANYTHING ELSE
```

```
00054   C       WILL SEND OUTPUT TO THE LINEPRINTER, DESIGNED FOR
00055   C       ACCESS BY ARMCO TERMINAL
00056   C
00057           CALL ASNLUN(1,'DP',IUNIT)
00058           CALL DPFILE(1,'FRMTTMP',4000,,IPNT1)
00059   C
00060           LUN1 = 6
00061           IF(INODE(200).EQ.5)LUN1 = 5
00062   C
00063   C       RETRIEVE THE DATA
00064   C
00065           RETRIEVE-THE-TEMPERATURE-DATA
00066   C
00067   C
00068   C *** WRITE HEADER INFORMATION TO THE PRINTER ***
00069   C
00070           WRITE(LUN1,500) WLDES,NPTS,CASELV
00071   500     FORMAT(45X,'TEMPERATURE DATA - WELL NO. ',12A1//
00072         1 50X,'MEASURMENTS TO DATE = ',I3/
00073         2 50X,'CASING ELEVATION    = ',F6.2/
00074         3 50X,'DEGREES CENTIGRADE'//)
00075   C
00076   C *** CONVERT THE DATE TO ASCII ***
00077   C
00078           DO (T = 1,NPTS)
00079           .  DT(T,3) = '-'
00080           .  DT(I,7) = '-'
00081           .  IYR =  T(I)/10000.
00082           .  IMON = T(I)/100.-IYR*100.
00083           .  WHEN (IMON .GT. 0)
00084           .  .  IDAY    = T(I)-IMON*100.-IYR*10000.
00085           .  .  DT(I,4) = AMON(1,IMON)
00086           .  .  DT(I,5) = AMON(2,IMON)
00087           .  .  DT(I,6) = AMON(3,IMON)
00088           .  .  ENCODE (2,501,YR) IYR
00089   501     .  .  FORMAT(I2)
00090           .  .  ENCODE (2,501,DAY) IDAY
00091           .  .  DT(I,1) = DAY(1)
00092           .  .  DT(I,2) = DAY(2)
00093           .  .  DT(I,8) = YR(1)
00094           .  .  DT(I,9) = YR(2)
00095           .  ...FIN
00096           .  ELSE
00097           .  .  DO (M=1,9) DT(I,M)=' '
00098           .  ...FIN
00099           ...FIN
00100   C
00101   C *** DUMP THE DATA TO THE PRINTER ***
00102   C
00103           WRITE(LUN1,502) ((DT(K,M),M=1,9),H(K),K=1,NPTS)
00104   502     FORMAT(5(5X,9A1,2X,F7.2,3X))
00105           CALL EXIT

        ------------------------------------

00106           TO RETRIEVE-THE-TEMPERATURE-DATA
```

```
00107   C       .
00108   C    IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
00109   C    NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
00110   C    IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
00111   C    NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
00112   C    INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
00113   C       .
00114           .  IDKIN=INODE(50)
00115           .  NPTSIN=INODE(51)
00116           .  IDKEX=INODE(52)
00117           .  NPTSEX=INODE(53)
00118           .  INALOC=INODE(54)
00119   C       .
00120   C    READ THE DATA IN THE INITIAL ALLOCATION
00121   C       .
00122           .  HADSI=IDKIN
00123           .  WRDI=NPTSIN*2.
00124           .  TADSI=HADSI+INALOC
00125           .  CALL DPR(IPNT1,HADSI,H,WRDI)
00126           .  CALL DPR(IPNT1,TADSI,T,WRDI)
00127   C       .
00128   C    READ THE EXTENDED ALLOCATION (IF ANY)
00129   C       .
00130           .  IF (NPTSEX .NE. 0)
00131           .  .  HADSE=IDKEX
00132           .  .  WRDE=NPTSEX*2.
00133   C       .  .
00134   C    FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
00135   C    TEMPERATURE AND TIME
00136   C       .  .
00137           .  .  TADSE=HADSE+2.
00138           .  .  NDX=NPTSIN+1
00139           .  .  CALL DPR(IPNT1,HADSE,H(NDX),WRDE)
00140           .  .  CALL DPR(IPNT1,TADSE,T(NDX),WRDE)
00141           .  ...FIN
00142   C       .
00143           .  NPTS=NPTSIN + NPTSEX
00144           ...FIN
00145          END
```

PROCEDURE CROSS-REFERENCE TABLE

```
00106 RETRIEVE-THE-TEMPERATURE-DATA
      00069
```

(FLECS VERSION 22.46)

TMPGLD.FLX

TEMPERATURE (GOULD)

```
00001   C ***************  [351,100]TMPGLD.FLX  ***************
00002   C
00003   C
00004   C  PROGRAM ID: SRV-11-WEL-10-3
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL TEMPERATURE DATA FROM THE
00008   C     CIRMIS DATA BANK AND DRAW A CAULD PLOT OF THE DATA.  THIS PROGRAM
00009   C     IS STARTED BY 'MNTR11' AND RECEIVES CONTROL DATA VIA THE TRANSFER
00010   C     NODE.
00011   C
00012   C  DATA FILES:
00013   C        NAME                        LUN              TYPE             ACCESS
00014   C        FILE 0---FRMTTMP             1               RAN               R
00015   C
00016   C
00017   C  LOADING SEQUENCE:
00018   C
00019   C     MCR>TKB @[351,100]TMPGLD
00020   C
00021   C
00022   C  BATTELLE MEMORIAL INSTITUTE
00023   C  PACIFIC NORTHWEST LABORATORIES
00024   C  WATER & LAND RESOURCES DEPT.
00025   C
00026   C  AUTHOR(S):   DR FRIEDRICHS
00027   C
00028   C  DATE:        INITIAL VERSION      JANUARY 1977
00029   C               CURRENT VERSION      JANUARY 1979
00030   C
00031   C
00032         BYTE A(3),B(2),WLDES
00033   C
00034         INTEGER BUFFER,SIZ,SIZ1,SIZ2,YC
00035         DIMENSION H(512),T(512),INODE(255),IBUF(253)
00036   C
00037         COMMON/BFR/ BUFFER(4800)
00038   C
00039         COMMON/HDR/ IDM(2),WLDES(12),IDM1(6),XXC,YYC,CASELV,IDM2(183),
00040        1            ICTM,NEX,MINX,MINY,MAXX,MAXY,MAPTYP,JPNT,
00041        2            NDEL,IDEL(40),IDISC,IUNIT,ICNTRL
00042   C
00043         EQUIVALENCE (WLDES(1),IBUF(1)), (IDM(1),INODE(1))
00044         EQUIVALENCE (IDKV,IBUF(55)),(NPTV,IBUF(56)),(IDTYP,IBUF(229))
00045   C
00046   C
00047   C *** READ THE TRANSFER NODE ***
00048   C
00049   C     TASK  =  RAD50('MNTR45')
00050         CALL VRECSP (TASK,INODE,255,,,IDS)
00051         JPNT  =  0
00052         NDEL  =  0
00053   C
```

```
00054          CALL ASNLUN(1,IDISC,IUNIT)
00055          CALL OPFILE(1,'FRMTTMP',4000,,IPNT1)
00056    C
00057    C *** RETRIEVE THE TEMPERATURE DATA ***
00058    C
00059    C
00060    C    IDKIN DISK ADDRESS OF THE INITIAL ALLOCATION
00061    C    NPTSIN NUMBER OF DATA POINTS IN THE INITIAL ALLOCATION
00062    C    IDKEX DISK ADDRESS OF THE EXTENDED ALLOCATION
00063    C    NPTSEX NUMBER OF DATA POINTS IN THE EXTENDED ALLOCATION
00064    C    INALOC NUMBER OF SECTORS IN THE INITIAL ALLOCATION
00065    C
00066          IDKIN = IBUF(50)
00067          NPTSIN = IBUF(51)
00068          IDKEX = IBUF(52)
00069          NPTSEX = IBUF(53)
00070          INALOC = IBUF(54)
00071    C
00072    C    READ THE DATA IN THE INITIAL ALLOCATION
00073          HADSI = IDKIN
00074          WRDI = NPTSIN*2,
00075          TADSI = HADSI+INALOC
00076          CALL OPR(IPNT1,HADSI,H,WRDI)
00077          CALL OPR(IPNT1,TADSI,T,WRDI)
00078    C
00079    C    READ THE EXTENDED ALLOCATION (IF ANY)
00080    C
00081          IF (NPTSEX .NE. 0)
00082          .  HADSE = IDKEX
00083          .  WRDE  = NPTSEX*2,
00084    C     .
00085    C    FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
00086    C    HEIGHT AND TIME
00087    C     .
00088          .  TADSE = HADSE+2,
00089          .  NDX   = NPTSIN+1
00090          .  CALL OPR(IPNT1,HADSE,H(NDX),WRDE)
00091          .  CALL OPR(IPNT1,TADSE,T(NDX),WRDE)
00092          ...FIN
00093          NTOT = NPTSIN + NPTSEX
00094    C
00095          IF (MINX .EQ. 0) NEX = 0
00096          CONDITIONAL
00097          .  (NEX .EQ. 0)
00098    C     .  .
00099    C *** CALCULATE AUTO SCALE FACTORS ***
00100    C     .  .
00101          .  .  TMAX = 0.0
00102          .  .  TMIN = 999999.
00103          .  .  HMAX = 0.0
00104          .  .  HMIN = 1.0E+20
00105          .  .  DO (I=1,NTOT)
00106          .  .  .  IF (T(I) .GT. TMAX) TMAX = T(I)
00107          .  .  .  IF (T(I) .LT. TMIN) TMIN = T(I)
00108          .  .  .  IF (H(I) .GT. 0.)
00109          .  .  .  .  IF (H(I) .GT. HMAX) HMAX = H(I)
```

```
00110          .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00111          .  .  .  ...FIN
00112          .  .  ...FIN
00113          .  .  MINX = IFIX(TMIN/10000.)
00114          .  .  MAXX = IFIX(TMAX/10000.)
00115          .  .  FIND=MINY=MAXY
00116          .  ...FIN
00117          .  (NEX .NE. 99)
00118          .  .  MINX = IBUF(232)
00119          .  .  MINY = IBUF(233)
00120          .  .  MAXX = IBUF(234)
00121          .  .  MAXY = IBUF(235)
00122          .  ...FIN
00123          .  (NEX .EQ. 99)
00124          .  .  MINX = IBUF(232)
00125          .  .  MAXX = IBUF(234)
00126          .  .  HMAX = 0.0
00127          .  .  HMIN = 1.0E+20
00128          .  .  DO (I=1,NTOT)
00129          .  .  .  NYR = T(I)/10000.
00130          .  .  .  IF (NYR .GE. IBUF(232) .AND. NYR .LE. IBUF(234))
00131          .  .  .  .  IF (H(I) .LT. HMIN) HMIN = H(I)
00132          .  .  .  .  IF( H(I) .GT. HMAX) HMAX = H(I)
00133          .  .  .  ...FIN
00134          .  .  ...FIN
00135          .  .  IF (HMAX .EQ. 0.) GO TO 46
00136          .  .  FIND=MINY=MAXY
00137          .  ...FIN
00138          ...FIN
00139    14    IMN=170
00140          IMX=770
00141          JMN=300
00142          JMX=900
00143          SCLX=IMX-IMN
00144          SCLY=JMX-JMN
00145          SIZ=24
00146          SIZ1=16
00147          SIZ2=8
00148    C---DEFINE BACKGROUND AND PLOT IT
00149          X1=FLOAT(IMN)
00150          X2=FLOAT(IMX)
00151          Y1=FLOAT(JMN)
00152          Y2=FLOAT(JMX)
00153    C
00154    C *** DRAW THE PLOT ***
00155    C
00156          CALL DOTINI(7,BUFFER,32,1200)
00157          DO 9999 INDEX=1,864,32
00158    C
00159          CALL FEDCL(0.0,0.0,0.0,1)
00160    C---DEFINE BACKGROUND AND PLOT IT
00161          CALL DOTLIN(IMN,JMN,IMN,JMX)
00162          CALL DOTLIN(IMN,JMX,IMX,JMX)
00163          CALL DOTLIN(IMX,JMX,IMX,JMN)
00164          CALL DOTLIN(IMX,JMN,IMN,JMN)
00165          TXS=120
```

```
00166          IYS=390
00167          CALL DOTCHR(IXS,IYS,'TEMPERATURE (DEGREES-C)',,SIZ,1)
00168          CALL DOTCHR(360,250,'CALENDAR YEAR',,SIZ)
00169          IX=360
00170          IY=150
00171          IZ=SIZ1
00172          CALL DOTCHR(IX,IY,'TEMPERATURE HISTORY',,IZ)
00173          IXL=(SCLX/2.) - FLOAT(SIZ2)*.75*16. + X1
00174          IY=130
00175          CALL DOTCHR(IXL,IY,'WELL DESIGNATION = ',,SIZ2)
00176          CALL DOTCHR(IXL+120,IY,WLDES,12,SIZ2)
00177          NDIF=MAXY-MINY
00178          YSCALE=SCLY/FLOAT(NDIF)
00179    C
00180    C
00181    C----DRAW Y LINES FOR CHART
00182          NN=MINY
00183          JDD=1
00184          NDIF=NDIF/INCY+1
00185          DO 3604 JD=1,NDIF
00186          Y= FLOAT(NN-MINY)*YSCALE+FLOAT(JMN)
00187          IY = Y
00188          IF(JDD.NE.1) GO TO 3602
00189          IF(JD.EQ.1.OR.JD.EQ.NDIF) GO TO 3601
00190          CALL DOTLIN (IMN,IY,IMX,IY)
00191    3601  ENCODE (3,321,A) NN
00192    321   FORMAT(I3)
00193          CALL DOTCHR(IMN-30,IY,A,3,,ISIZ2)
00194          GO TO 3603
00195    3602  CALL DOTLIN (IMN,IY,IMN+10,IY)
00196    3603  JDD=JDD+1
00197          IF(JDD.EQ.6) JDD=1
00198          NN=NN+INCY
00199    3604  CONTINUE
00200    C
00201    C
00202    C----DRAW X LINES FOR CHART
00203          NDIF=(MAXX-MINX) + 1
00204          XSCALE=SCLX/(FLOAT(NDIF)*365.)
00205          XINC=(SCLX/FLOAT(NDIF))
00206          XADJ=XINC/2. + FLOAT(SIZ2)*.75
00207          JDD=1
00208          DO 3650 JD=MINX,MAXX
00209          X=FLOAT(JD-MINX+1)*XINC + X1
00210          IX=X-XADJ
00211          ENCODE(2,636,B) JD
00212    636   FORMAT(I2)
00213          CALL DOTCHR(IX,JMN-20,B,2,SIZ2)
00214          IX=X
00215          IF(JD.EQ.MAXX) GO TO 3650
00216          GO TO(3620,3630),JDD
00217    3620  JDD=2
00218          CALL DOTLIN(IX,JMN,IX,JMN+10)
00219          GO TO 3650
00220    3630  JDD=1
00221          CALL DOTLIN(IX,JMN,IX,JMX)
```

```
00222    3650  CONTINUE
00223    C
00224    C
00225    C---DRAW THE TEMPERATURE PROFILE
00226    C
00227          IF(INDEX.EQ.1) WRITE(6,637) NTOT,(T(I),H(I),I=1,NTOT)
00228    637   FORMAT(' NTOT= ',I3,/,(1X,2F10.2))
00229          NABSV=0
00230          DO 40 JK=1,NTOT
00231    3701  Y=H(JK)
00232          IYR=T(JK)/10000.
00233          MO=T(JK)/100.-FLOAT(IYR)*100.
00234          IDA=T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
00235    38    DAYS=(FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
00236          X=(DAYS*XSCALE+FLOAT(IMN))
00237          Y = (Y-FLOAT(MINY))*YSCALE + JMN
00238          IX = X
00239          IY = Y
00240    C     CHECK FOR POINT WITHIN REGION
00241    3806  IF(X.LT.X1.OR.X.GT.X2) GO TO 3850
00242          IF(Y.GE.Y1.AND.Y.LE.Y2) GO TO 39
00243    3850  NABSV = 0
00244          GO TO 3990
00245    C  YES---PLOT DATA POINT
00246    39    IF (NABSV.EQ.0) GO TO 3980
00247          CALL DOTLIN (LASTX,LASTY,IX,IY)
00248    3980  NABSV = 1
00249    3990  LASTX = IX
00250          LASTY = IY
00251    40    CONTINUE
00252    C
00253    C  PLOT DATA POINTS
00254    9990  CALL DOTOUT
00255          CALL DOTDUN
00256    46    IF (ICNTRL. EQ .1)
00257          .  IEFN = 33
00258          .  CALL SETEF(IEFN,IOS)
00259          ...FIN
00260          CALL EXIT
```

```
00261          TO FIND-MINY-MAXY
00262          .  MINY = IFIX(HMIN/10.)*10-10
00263          .  MAXY = IFIX(HMAX/10.)*10+10
00264          .  INCY = 10
00265          .  IF ((MAXY-MINY). LE .100) INCY = 5
00266          .  IF ((MAXY-MINY). LE .40) INCY = 2
00267          ...FIN
00268          END
```

PROCEDURE CROSS-REFERENCE TABLE

00261 FIND-MINY-MAXY

00115 00136

(FLECS VERSION 22.46)

```
00269    C
00270    C
00271          SUBROUTINE FEDCL(XX,YY,IR)
00272    C
00273          INTEGER X,Y,BUFFER
00274    C
00275          COMMON/HER/ BUFFER(4800)
00276    C
00277    C
00278          XMAX=850.
00279          YMAX=1100.
00280          IF(IR.EQ.1) GO TO 10
00281          XMAX=1100.
00282          YMAX=850.
00283    10    Y=YY
00284          DO 15 J=1,4
00285          X=XX
00286          IF(J.GE.3) Y=YY+YMAX
00287          IF(J.EQ.2.OR.J.EQ.4) X=XX+XMAX
00288          CALL DOTLIN(X,Y+10,X+20,Y+10)
00289          CALL DOTLIN(X+10,Y,X+10,Y+20)
00290    15    CONTINUE
00291          RETURN
00292          END
```

(FLECS VERSION 22.46)

CIRMIS.FLX

11/55 CIRMIS MONITOR

```
00001    C ******************** [351,101]CIRMIS.FLX ********************
00002    C
00003    C
00004    C  PROGRAM ID: MON-55  (11/55 CIRMIS MONITOR)
00005    C
00006    C  PROGRAM DESCRIPTION:
00007    C     THIS PROGRAM IS THE MAIN CONTROLER FOR THE CIRMIS SYSTEM.
00008    C
00009    C  DATA FILES:
00010    C        NAME                        LUN          TYPE          ACCESS
00011    C        FILE0(DK1)---WELSPEC         4           RAN            R/W
00012    C
00013    C
00014    C
00015    C
00016    C  BATTELLE MEMORIAL INSTITUTE
00017    C  PACIFIC NORTHWEST LABORATORIES
00018    C  WATER & LAND RESOURCES DEPT.
00019    C
00020    C  AUTHOR(S):   DR FRIEDRICHS
00021    C
00022    C  DATE:  INTIIAL VERSION  FEBRUARY 4, 1978
00023    C         CURRENT VERSION  FEBRUARY 21, 1978
00024    C
00025    C
00026         DIMENSION MAIN(100),LABEL(200),INODE(262)
00027    C
00028         CALL MAINOF(MAIN,100,0,1)
00029         CALL CLKOFF
00030    C
00031         CALL SUBOF (LABEL,200,0)
00032         CALL INTENS (7)
00033         CALL CHRSCL (2)
00034         CALL APOS (145,895)
00035         CALL TEXT ('COMPREHENSIVE INFORMATION RETRIEVAL',-30)
00036         CALL TEXT ('    AND MODEL INPUT SEQUENCE',-30)
00037         CALL RPOS (252,0)
00038         CALL CHRTYP (1,0,0)
00039         CALL TEXT ('(CIRMIS)')
00040    C
00041         CALL APOS (100,800)
00042         CALL BOX (820,150)
00043         CALL INTENS (3)
00044         CALL RPOS (5,5)
00045         CALL BOX (810,140)
00046    C
00047         CALL CHRSCL (1)
00048         CALL APOS (310,600)
00049         CALL TEXT ('SELECT A FUNCTIONAL CATEGORY:',-3)
00050         CALL RVEC (400,0)
00051         CALL LPHIT (1)
00052         IX = 310
00053         IY = 550
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:07  PAGE  00002
  00054          DO (J=1,3)
  00055          .  CALL NAME (J)
  00056          .  CALL APOS (IX,IY)
  00057          .  SELECT (J)
  00058          .  .  (1)
  00059          .  .  .  CALL TEXT ('WELL BASED DATA RETRIEVAL')
  00060          .  .  ...FIN
  00061          .  .  (2)
  00062          .  .  .  CALL TEXT ('SPECIAL APPLICATIONS AND MANIPULATIONS')
  00063          .  .  ...FIN
  00064          .  .  (3)
  00065          .  .  .  CALL TEXT ('MODEL INPUT SEQUENCING ROUTINES')
  00066          .  .  ...FIN
  00067          .  ...FIN
  00068          .  IY = IY-50
  00069          ...FIN
  00070          CALL CHRSCL (0)
  00071          CALL NAME (4)
  00072          CALL APOS (700,100)
  00073          CALL TEXT ('EXIT CIRMIS')
  00074          CALL LPHIT (0)
  00075          CALL ENDSR (ISZ)
  00076    C     WRITE(5,509) ISZ
  00077    509   FORMAT(' ISZ = ',I5)
  00078    C
  00079    C
  00080    C*******************************************************************************
  00081    C
  00082    C     START PROGRAM
  00083    C
  00084    C*******************************************************************************
  00085    C
  00086    C
  00087    10    CALL GRATIN (1,ITYP,'LP')
  00088    C
  00089          IF (ITYP.NE.'LP') GO TO 10
  00090          CALL LTPEN (ID,ITIP)
  00091          IF (ITIP.EQ.1)
  00092          .  IF (ID.EQ.4) STOP
  00093          .  IF (ID.GE.1.AND.ID.LE.3)
  00094          .  .  SELECT (ID)
  00095          .  .  .  (1)
  00096          .  .  .  .  DO (J=1,262) INODE(J)=0
  00097          .  .  .  .  CALL DPFILE (4,'WELSPEC',2,,IPNT0)
  00098          .  .  .  .  CALL DPW (IPNT0,0,,INODE,262,)
  00099          .  .  .  .  TASK = RAD50('WELSEL')
  00100          .  .  .  ...FIN
  00101          .  .  .  (2) TASK = RAD50('APPLIC')
  00102          .  .  .  (3) TASK = RAD50('MODINP')
  00103          .  .  ...FIN
  00104          .  .  CALL REQUES (TASK,,IDS)
  00105          .  .  IF (IDS.GT.0) CALL EXIT
  00106          .  .  WRITE(5,530) IDS
  00107    530   .  .  FORMAT(' REQUEST ERROR.  IDS = ',I5)
  00108          .  .  III = 0
  00109          .  .  IF (III.EQ.0) STOP


(FLECS VERSION 22.46)     14-SEP-79   07:58:07  PAGE  00003
  00110          .  ...FIN
  00111          ...FIN
  00112          GO TO 10
  00113          END

          (FLECS VERSION 22.46)

          ------------------------------------
```

WELSEL.FLX

WELL BASED DATA SELECTOR

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00001

        ----------------------------------------

00001   C ********************  [351,101]WELSEL.FLX  ********************
00002   C
00003   C
00004   C  PROGRAM ID:  SEL-55-WEL
00005   C
00006   C  PROGRAM DESCRIPTION:  THIS PROGRAM IS THE MAIN CONTROLLER FOR
00007   C                        SELECTING AND OUTPUTTING WELL BASED DATA.
00008   C
00009   C
00010   C  DATA FILES:
00011   C       NAME                      LUN          TYPE          ACCESS
00012   C       ----                      ---          ----          ------
00013   C
00014   C       VS60---DISPLAY             1           ---            R/W
00015   C       VS60---KEYBOARD            2           ---             R
00016   C       FILEQ(UB0)---WELLHDR       3           RAN             R
00017   C       FILEQ(DB0)---WELLNAM       3           RAN             R
00018   C       FILEQ(DK1)---WELSPEC       4           RAN            R/W
00019   C       NT------DECNET             6           ---            R/W
00020   C
00021   C
00022   C
00023   C
00024   C  BATTELLE MEMORIAL INSTITUTE
00025   C  PACIFIC NORTHWEST LABORATORIES
00026   C  WATER & LAND RESOURCES DEPT.
00027   C
00028   C  AUTHOR(S):   DR FRIEDRICHS
00029   C
00030   C  DATE:  INITIAL VERSION  FEBRUARY 4, 1978
00031   C         CURRENT VERSION  AUGUST  28, 1979
00032   C
00033   C
00034   C
00035         BYTE WLDES,WNAME,KEYBUF(80)
00036         INTEGER CTMSEL,WORKSP(35),PASSWD(4),STATUS(2),UIC(2)
00037         LOGICAL FOUND, BUSY
00038   C
00039         DIMENSION IBIT(5),ISZ(12),IBUF(280),MSGBUF(20)
00040         DIMENSION MATRIX(1345),LABEL(45),
00041        1      ISPECS(80),LBUT1(70),MESS1(750),NXTPG(35),NOWEL(20),
00042        2      LCNTRL(25),LPAGE2(25),MTxSEL(160),NAMCTM(560),
00043        3      NXTPG1(35),NRTRV(40),NAMDSP(20),CTMSEL(200),LPAGE1(25),
00044        4      IOUT(100),MESS2(55),ISTAR(20)
00045   C
00046         COMMON/DSP/ MAIN(100),KEYOUT(100)
00047         COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IOBIT,ICBIT(2),
00048        1            IDM(234),IWNAM,IWHDR,NRETRN,MODE,WNAME(12),IOTYP,
00049        2            IDTYP,ICTYP,NVAR,IVAR(11)
00050   C
00051         EXTERNAL TRAP
00052         EQUIVALENCE (IBUF(1),WLDES(1)),(MSGBUF(1),WNAME(1))
00053   C
```

```
(FLECS VERSION 22.461     14-SEP-79   07:58:22  PAGE  00002
  00054          DATA UTC /0351,0100/
  00055    C
  00056          CALL MAINDF (MAIN,100,0,1,0,2)
  00057          CALL CLKOFF
  00058    C
  00059    C
  00060          JMTX = 1
  00061          JNCTM = 1
  00062          JCSEL = 1
  00063          II2 = 1
  00064          IT3 = 1
  00065          IST = 3
  00066          IBIT(1) = "2057
  00067          IBIT(2) = "2057
  00068          IBIT(3) = "2057
  00069          IBIT(4) = "13
  00070          IBIT(5) = "10
  00071          NVAR = 0
  00072    C
  00073    C
  00074          SETUP-DISPLAY-FILES
  00075          SETUP-MATRIX-DISPLAY
  00076          SETUP-CONTAMINANT-NAMES-DISPLAY
  00077          CALL DPFILE (4,'WELSPEC',2,,IPNT0)
  00078          CALL DPR (IPNT0,0,,IBUF,280,)
  00079          CALL STRTSB (LABEL)
  00080          CONDITIONAL
  00081          .  (NRETRN.EQ.0)
  00082          .  .  CALL STRTSB (LBUT1)
  00083          .  .  IOTYP = 1
  00084          .  .  IOTYP = 1
  00085          .  ...FIN
  00086          .  (NRETRN.EQ.1, OR .NRETRN.EQ.2)
  00087          .  .  DO (J=1,12) WNAME(J)=WLDES(J)
  00088          .  .  SETUP-WELL-NAME-DISPLAY
  00089          .  .  SETUP-WELL-PARAMETERS-DISPLAY
  00090          .  .  SETUP-MATRIX-SELECT-BUTTONS
  00091          .  .  CALL STRTSB (NRTRV,LPAGE2,LCNTRL,IOUT,ISPECS)
  00092          .  .  SELECT (NRETRN)
  00093          .  .  .  (1) CALL STRTSB (MATRIX,MTXSEL)
  00094          .  .  .  (2)
  00095          .  .  .  .  SETUP-CONTAMINANT-LIGHT-BUTTONS
  00096          .  .  .  .  CALL STRTSB (NAMCTM,LPAGE1)
  00097          .  .  .  ...FIN
  00098          .  .  ...FIN
  00099          .  ...FIN
  00100          ...FIN
  00101          IKB = 0
  00102          BUSY = .FALSE.
  00103          CALL DPFILE (3,'WELLNAM',101,,IWNAM)
  00104          CALL DPFILE (3,'WELLHDR',6000,,IWHDR)
  00105    C
  00106    19    CALL GRATIN (1,ITYP,'LP','KB')
  00107          SELECT (ITYP)
  00108    C *** KEYBOARD INTERRUPT ***
  00109          .  ('KB')
```

```
(FLECS VERSION 22.46)     14-SEP-79   07158122  PAGE  00003
  00110          .  .  CALL STOPSB (NAMDSP)
  00111          .  .  DO (J=1,12) KEYBUF(J) = ' '
  00112          .  .  CALL GRECHO (KEYBUF,NCHAR,380,600,2,1,IKB)
  00113          .  .  DECODE (NCHAR,220,KEYBUF) WNAME
  00114   220    .  .  FORMAT (12A1)
  00115          .  .  SETUP-WELL-NAME-DISPLAY
  00116          .  .  CALL STRTSB (NAMDSP)
  00117          .  .  SEE-IF-WELL-EXISTS
  00118          .  .  WHEN (FOUND)
  00119          .  .  .  SETUP-WELL-PARAMETERS-DISPLAY
  00120          .  .  .  CALL STRTSB (NXTPG,IOUT,ISPECS)
  00121          .  .  ...FIN
  00122          .  .  ELSE
  00123          .  .  .  CALL STRTSB (NOWEL)
  00124          .  .  .  CALL WAIT (5,2,MMM)
  00125          .  .  .  CALL STOPSB (NOWEL)
  00126          .  .  ...FIN
  00127          .  ...FIN
  00128   C *** LIGHT PEN INTERRUPTS ***
  00129          .  ('LP')
  00130          .  .  CALL LTPEN (ID,ITIP)
  00131          .  .  IF(ITIP.EQ.1)
  00132          .  .  .  IF (.NOT.BUSY)
  00133          .  .  .  .  BUSY = .TRUE.
  00134          .  .  .  .  CALL WAIT (1,2,MMM)
  00135          .  .  .  .  IF (ID.GE.1. AND .ID.LE.2)
  00136          .  .  .  .  .  CALL SUBDF (ISTAR,20,IST)
  00137          .  .  .  .  .  IST = 2
  00138          .  .  .  .  .  CALL INTENS (4)
  00139          .  .  .  .  .  CALL CHRSCL (1)
  00140          .  .  .  .  .  IY = 620 - (ID-1)*80
  00141          .  .  .  .  .  CALL APOS (350,IY)
  00142          .  .  .  .  .  CALL TEXT ('*')
  00143          .  .  .  .  .  CALL ENDSB
  00144          .  .  .  .  .  CALL WAIT (1,2,MMM)
  00145          .  .  .  .  ...FIN
  00146          .  .  .  .  CONDITIONAL
  00147          .  .  .  .  .  (ID.GE.1. AND .ID.LE.2)
  00148          .  .  .  .  .  .  SELECT (ID)
  00149   C ** INPUT WELL DESIGNATION FROM SCOPE KEYBOARD
  00150          .  .  .  .  .  .  .  .  (1)
  00151          .  .  .  .  .  .  .  .  .  MODE = 1
  00152          .  .  .  .  .  .  .  .  .  CALL STOPSB (LBUT1)
  00153          .  .  .  .  .  .  .  .  .  CALL STRTSB (MESS1,LCNTRL)
  00154          .  .  .  .  .  .  .  .  .  CALL SUBDF (KEYOUT,100,IKB)
  00155          .  .  .  .  .  .  .  .  .  IKB = 2
  00156          .  .  .  .  .  .  .  .  .  CALL INTENS (4)
  00157          .  .  .  .  .  .  .  .  .  CALL CHRSCL (2)
  00158          .  .  .  .  .  .  .  .  .  CALL APOS (382,600)
  00159          .  .  .  .  .  .  .  .  .  CALL TEXT ('>SE')
  00160          .  .  .  .  .  .  .  .  .  CALL ENDSB
  00161          .  .  .  .  .  .  .  .  .  CALL STRTSB (KEYOUT)
  00162          .  .  .  .  .  .  .  .  ...FIN
  00163   C ** SELECT WELL FROM MAP LOCATION
  00164          .  .  .  .  .  .  .  .  (2)
  00165          .  .  .  .  .  .  .  .  .  DISPLAY-WELL-LOCATION-MAP
```

```
(FLECS VERSION 22.46)     14-SEP-79   07158122  PAGE  00004
  00166          .  .  .  .  .  .  .  ...FIN
  00167    C ** SELECT WELL FROM PAGED LISTING
  00168          .  .  .  .  .  .  ...FIN
  00169          .  .  .  .  .  .  CALL STOPSB (ISTAR)
  00170          .  .  .  .  .  ...FIN
  00171    C  ** START DATA TYPE SELECT DISPLAY
  00172          .  .  .  .  (ID.EQ.4)
  00173          .  .  .  .  .  CALL STOPSB (MESS1,NXTPG,KEYOUT,NAMDSP,NRTRV)
  00174          .  .  .  .  .  CALL STRTSB (MATRIX,LPAGE2)
  00175          .  .  .  .  .  SETUP-MATRIX-SELECT-BUTTONS
  00176          .  .  .  .  .  CALL STRTSB (MTXSEL)
  00177          .  .  .  .  ...FIN
  00178    C  ** RETURN TO CIRMIS MONITOR
  00179          .  .  .  .  (ID.EQ.5)
  00180          .  .  .  .  .  CALL CLROEV ('KB')
  00181          .  .  .  .  .  CALL WAIT (2,2,MMM)
  00182          .  .  .  .  .  NRETRN = 0
  00183          .  .  .  .  .  CALL DPW (IPNT0,0,,IBUF,258,)
  00184          .  .  .  .  .  TASK = RAD50 ('CIRMI8')
  00185          .  .  .  .  .  CALL REQUES (TASK,,ID8)
  00186          .  .  .  .  .  IF (ID.EQ.5) CALL EXIT
  00187          .  .  .  .  ...FIN
  00188    C  ** RETURN TO WELL SELECT DISPLAY
  00189          .  .  .  .  (ID.EQ.6)
  00190          .  .  .  .  .  SELECT (MODE)
  00191          .  .  .  .  .  .  (1)
  00192          .  .  .  .  .  .  .  CALL STOPSB (MATRIX,MTXSEL,NAMCTM,LPAGE2,CTMSEL,NRTRV)
  00193          .  .  .  .  .  .  .  CALL STRTSB (MESS1,LCNTRL,NAMDSP,LPAGE1)
  00194          .  .  .  .  .  .  ...FIN
  00195          .  .  .  .  .  .  (2)
  00196          .  .  .  .  .  .  .  DISPLAY-WELL-LOCATION-MAP
  00197          .  .  .  .  .  .  ...FIN
  00198          .  .  .  .  .  ...FIN
  00199          .  .  .  .  ...FIN
  00200    C  ** RETURN TO DATA SELECT PAGE
  00201          .  .  .  .  (ID.EQ.7)
  00202          .  .  .  .  .  CALL STOPSB (MESS1,NXTPG,KEYOUT,NAMCTM,LPAGE1,CTMSEL)
  00203          .  .  .  .  .  CALL STOPSB (NRTRV,NAMDSP,NXTPG1)
  00204          .  .  .  .  .  CALL STRTSB (MATRIX,LPAGE2,MTXSEL)
  00205          .  .  .  .  ...FIN
  00206    C  ** DISPLAY CONTAMINANT LIGHT BUTTONS
  00207          .  .  .  .  (ID.EQ.10)
  00208          .  .  .  .  .  CALL STOPSB (NXTPG1,MATRIX,MTXSEL)
  00209          .  .  .  .  .  CALL STRTSB (NAMCTM,LPAGE1,NRTRV)
  00210          .  .  .  .  .  SETUP-CONTAMINANT-LIGHT-BUTTONS
  00211          .  .  .  .  ...FIN
  00212    C ** RETRIEVE DATA AND START PROGRAM
  00213          .  .  .  .  (ID.EQ.11)
  00214          .  .  .  .  .  CALL STOPSB (NRTRV)
  00215          .  .  .  .  .  CALL STRTSB (MESS2)
  00216          .  .  .  .  .  NVAR = 0
  00217          .  .  .  .  .  WHEN (IOTYP.NE.0)
  00218          .  .  .  .  .  .  ICON = 0
  00219          .  .  .  .  .  .  PASSWD(1) = 351
  00220          .  .  .  .  .  .  CALL NTINIT (STATUS,35,WORKSP)
  00221          .  .  .  .  .  .  CALL NTCONW (6,STATUS,ICON,'WLR45_','MNTR45',,UIC,1,PASSWD,TRAP)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  0005
  00222                 .  .  .  .  .  .  .  CALL NTSNDW (6,STATUS,40,MSGBUF)
  00223                 .  .  .  .  .  .  .  CONDITIONAL
  00224                 .  .  .  .  .  .  .  .  (STATUS(1).LT.1)
  00225                 .  .  .  .  .  .  .  .  .  WRITE(5,535) STATUS(1)
  00226   535           .  .  .  .  .  .  .  .  .  FORMAT(' TARGET IS NOT RECEPTIVE, STATUS =',I5)
  00227                 .  .  .  .  .  .  .  .  .  CALL NTWAIT (6,STATUS)
  00228                 .  .  .  .  .  .  .  .  ...FIN
  00229                 .  .  .  .  .  .  .  .  (STATUS(1).EQ.1)
  00230                 .  .  .  .  .  .  .  .  .  CALL NTDISW (6,STATUS,,)
  00231                 .  .  .  .  .  .  .  .  ...FIN
  00232                 .  .  .  .  .  .  .  .  (STATUS(1).GT.1)
  00233                 .  .  .  .  .  .  .  .  .  WRITE(5,536) STATUS(1)
  00234   536           .  .  .  .  .  .  .  .  .  FORMAT(' ERROR DURING COMMUNICATION--CODE=',I5)
  00235                 .  .  .  .  .  .  .  .  ...FIN
  00236                 .  .  .  .  .  .  .  ...FIN
  00237                 .  .  .  .  .  .  .  CALL NTDISW (6,STATUS)
  00238                 .  .  .  .  .  .  .  CALL WAIT (2,2,MMM)
  00239                 .  .  .  .  .  .  .  CALL STOPSB (MESS2)
  00240                 .  .  .  .  .  .  .  CALL STRTSB (NRTRV)
  00241                 .  .  .  .  .  .  ...FIN
  00242                 .  .  .  .  .  .  ELSE
  00243                 .  .  .  .  .  .  .  CALL DPW (IPNT0,0,,IBUF,280,)
  00244                 .  .  .  .  .  .  .  CALL CLRDEV ('K8')
  00245                 .  .  .  .  .  .  .  CALL WAIT (2,2,MMM)
  00246                 .  .  .  .  .  .  .  SELECT (IDTYP)
  00247                 .  .  .  .  .  .  .  .  (0) TASK = RAD50 ('HYDSCP')
  00248                 .  .  .  .  .  .  .  .  (1) TASK = RAD50 ('CTMSCP')
  00249                 .  .  .  .  .  .  .  .  (3) TASK = RAD50 ('WLGSCP')
  00250                 .  .  .  .  .  .  .  .  (10)TASK = RAD50 ('TMPSCP')
  00251                 .  .  .  .  .  .  .  ...FIN
  00252                 .  .  .  .  .  .  .  CALL REQUES (TASK,,IOS)
  00253                 .  .  .  .  .  .  .  IF(IOS.GT.0) CALL EXIT
  00254                 .  .  .  .  .  .  .  WRITE(5,548) IOS
  00255   548           .  .  .  .  .  .  .  FORMAT(' REQUEST ERROR,  IOS = ',I5)
  00256                 .  .  .  .  .  .  .  III = 0
  00257                 .  .  .  .  .  .  .  IF (III.EQ.0) STOP
  00258                 .  .  .  .  .  .  ...FIN
  00259                 .  .  .  .  .  ...FIN
  00260   C  ** CHECK FOR INTERRUPT ON DATA SELECT MATRIX
  00261                 .  .  .  .  (ID.GE.100. AND .ID.LE.204)
  00262                 .  .  .  .  .  IDTYP = ID/10 * 10
  00263                 .  .  .  .  .  IDTYP = ID - 100 -IDTYP*10
  00264                 .  .  .  .  .  SETUP-MATRIX-SELECT-BUTTONS
  00265                 .  .  .  .  .  CALL STRTSB (MTXSEL)
  00266                 .  .  .  .  .  CALL STOPSB (NXTPG1,NRTRV)
  00267                 .  .  .  .  .  WHEN (IDTYP.EQ.1) CALL STRTSB (NXTPG1)
  00268                 .  .  .  .  .  ELSE
  00269                 .  .  .  .  .  .  CALL STRTSB (NRTRV)
  00270                 .  .  .  .  .  ...FIN
  00271                 .  .  .  .  ...FIN
  00272   C  ** CHECK FOR INTERRUPT ON CONTAMINANT NAME
  00273                 .  .  .  .  (ID.GE.901. AND .ID.LE.932)
  00274                 .  .  .  .  .  ICTYP = ID-900
  00275                 .  .  .  .  .  SETUP-CONTAMINANT-LIGHT-BUTTONS
  00276                 .  .  .  .  ...FIN
  00277                 .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   07158122  PAGE  00006
  00278          .  .  .  .  BUSY = .FALSE.
  00279          .  .  .  ...FIN
  00280          .  .  ...FIN
  00281          .  ...FIN
  00282          ...FIN
  00283          GO TO 19

         ----------------------------------------

  00284          TO DISPLAY-WELL-LOCATION-MAP
  00285          .  MODE = 2
  00286          .  CALL DPW (IPNT0,0,,IBUF,200,)
  00287          .  CALL CLRDEV ('KB')
  00288          .  CALL WAIT(2,2,MMM)
  00289          .  TASK = RAD50 ('DSPSEL')
  00290          .  CALL REQUES (TASK,,IOS)
  00291          .  IF (IOS.GT.0) CALL EXIT
  00292          ...FIN

         ----------------------------------------

  00293          TO SEE-IF-WELL-EXISTS
  00294          .  CALL RTVHDR (WNAME,FQADR)
  00295          .  FOUND=.TRUE.
  00296          .  IF (FQADR.LT.0.) FOUND=.FALSE.
  00297          .  IDTYP = "377
  00298          .  IDTYP = "377
  00299          .  ICTYP = 0
  00300          ...FIN

         ----------------------------------------

  00301          TO SETUP-WELL-NAME-DISPLAY
  00302          .  CALL SUBDF (NAMDSP,20,II2)
  00303          .  II2 = 3
  00304          .  CALL INTENS (4)
  00305          .  CALL CHRSCL (2)
  00306          .  CALL APOS (380,600)
  00307          .  CALL TEXT (WNAME,0,12)
  00308          .  CALL ENDSB
  00309          ...FIN

         ----------------------------------------

  00310          TO SETUP-WELL-PARAMETERS-DISPLAY
  00311          .  CALL SUBDF (IOUT,100,II3)
  00312          .  II3 = 3
  00313          .  CALL AREA (1)
  00314          .  CALL INTENS (4)
  00315          .  CALL CHRSCL (0)
  00316          .  CALL APOS (110,80)
  00317          .  CALL TEXT (WLDES,-15,12)
  00318          .  CALL FNMBR (CASEL,'(F7.2)',-15)
  00319          .  CALL FNMBR (XC,'(F7.0)',-15)
  00320          .  CALL FNMBR (YC,'(F7.0)',-15)
  00321          .  CALL AREA (0)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00007
  00322          . CALL END88
  00323          ...FIN

         ------------------------------------

  00324          TO SETUP-DISPLAY-FILE8
  00325          . CALL SUBDF (LABEL,45,1)
  00326          . CALL INTENS (7)
  00327          . CALL CHRSCL (2)
  00328          . CALL CHRTYP (1,0,0)
  00329          . CALL APOS (165,950)
  00330          . CALL TEXT ('WELL BASED DATA SELECTION ROUTINE')
  00331          . CALL APOS (130,915)
  00332          . CALL BOX (760,90)
  00333          . CALL INTENS (2)
  00334          . CALL RPOS (5,5)
  00335          . CALL BOX (750,80)
  00336          . CALL ENDSR(ISZ(1))
  00337    C     .
  00338          . CALL SUBDF (ISPECS,80,1)
  00339          . CALL AREA (1)
  00340          . CALL INTENS (4)
  00341          . CALL CHRSCL (0)
  00342          . CALL APOS (0,100)
  00343          . CALL TEXT ('CURRENTLY SELECTED WELL DATA:',-3)
  00344          . CALL RVEC (200,0)
  00345          . CALL APOS (0,80)
  00346          . CALL TEXT ('DESIGNATION  =',-15)
  00347          . CALL TEXT ('CASING ELEV  =',-15)
  00348          . CALL TEXT ('X COORDINATE =',-15)
  00349          . CALL TEXT ('Y COORDINATE =',-15)
  00350          . CALL AREA (0)
  00351          . CALL ENDSR(ISZ(2))
  00352    C     .
  00353    C *** WELL SELECT MODE ***
  00354    C     .                                          ID=1,2,3
  00355          . CALL SUBDF (LBUT1,70,1)
  00356          . CALL INTENS (4)
  00357          . CALL CHRSCL (1)
  00358          . CALL APOS (275,700)
  00359          . CALL TEXT ('CHOOSE A METHOD OF WELL SELECTION:',-3)
  00360          . CALL RVEC (475,0)
  00361          . IY = 620
  00362          . CALL LPHIT(1)
  00363          . DO (J=1,2)
  00364          . . CALL NAME (J)
  00365          . . CALL APOS (370,IY)
  00366          . . SELECT (J)
  00367          . . . (1) CALL TEXT ('KEYBOARD ENTRY')
  00368          . . . (2) CALL TEXT ('LOCATION MAP DISPLAY')
  00369          . . ...FIN
  00370          . . IY = IY-80
  00371          . ...FIN
  00372          . CALL LPHIT(0)
  00373          . CALL ENDSR(ISZ(3))
  00374    C     .
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00008
  00375          .  CALL SUBDF (MESS1,750,1)
  00376          .  CALL INTENS (4)
  00377          .  CALL CHRSCL (1)
  00378          .  CALL APOS (175,700)
  00379          .  CALL TEXT ('ENTER A WELL DESIGNATION FROM THE SCOPE KEYBOARD:',-3)
  00380          .  CALL RVEC (670,0)
  00381          .  ISPACE = 21
  00382          .  IX = 375
  00383          .  IY = 600
  00384          .  SETUP-FORMAT-SPACING-LINE
  00385          .  CALL CHRSCL (0)
  00386          .  CALL INTENS (5)
  00387          .  CALL APOS (375,570)
  00388          .  CALL TEXT ('(EXAMPLE FORMATS FOR EACH WELL CATEGORY',-15)
  00389          .  CALL TEXT (' ARE SHOWN IN THE BOX BELOW.)')
  00390          .  CALL CHRSCL (0)
  00391          .  CALL APOS (310,253)
  00392          .  CALL TEXT ('FORMATTING STRUCTURE FOR EACH OF THE SEVEN WELL')
  00393          .  CALL TEXT (' CATAGORIES')
  00394          .  CALL INTENS (7)
  00395          .  CALL APOS (300,75)
  00396          .  CALL BOX (420,195)
  00397          .  CALL INTENS (2)
  00398          .  CALL APOS (305,80)
  00399          .  CALL BOX (410,165)
  00400          .  CALL INTENS (4)
  00401          .  CALL APOS (440,230)
  00402          .  CALL TEXT ('  1      B 3        1  ',-20)
  00403          .  CALL TEXT ('2 E      2 7    1 1 5  ',-20)
  00404          .  CALL TEXT ('2 W      1 1      1 3 Q',-20)
  00405          .  CALL TEXT ('  3        3      1 0  ',-20)
  00406          .  CALL TEXT ('  6    S 5 8    E 3 A  ',-20)
  00407          .  CALL TEXT ('1 1      2 9    1 2 B  ',-20)
  00408          .  CALL TEXT ('3 0      4 1    1 3 C  ')
  00409          .  ISPACE = 14
  00410          .  IX = 437
  00411          .  IY = 235
  00412          .  DO (K=1,7)
  00413          .  .  SETUP-FORMAT-SPACING-LINE
  00414          .  .  IY = IY-20
  00415          .  ...FIN
  00416          .  CALL ENDSR(ISZ(4))
  00417    C     .
  00418    C     .                                              ID=4
  00419          .  CALL SUBDF (NXTPG,35,1)
  00420          .  CALL INTENS (6)
  00421          .  CALL CHRSCL (0)
  00422          .  CALL LPHIT(1)
  00423          .  CALL NAME (4)
  00424          .  CALL APOS (485,450)
  00425          .  CALL TEXT ('CONTINUE')
  00426          .  CALL LPHIT(0)
  00427          .  CALL APOS (475,440)
  00428          .  CALL BOX (75,27)
  00429          .  CALL INTENS (1)
  00430          .  CALL APOS (478,443)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07158122  PAGE  00009
  00431          .  CALL BOX (69,21)
  00432          .  CALL ENDSB(ISZ(5))
  00433   C      .
  00434          .  CALL SUBDF (NOWEL,20,1)
  00435          .  CALL INTENS (4)
  00436          .  CALL CHRSCL (1)
  00437          .  CALL BLINK (1)
  00438          .  CALL APOS (460,450)
  00439          .  CALL TEXT ('WELL NOT FOUND')
  00440          .  CALL BLINK (0)
  00441          .  CALL ENDSB(ISZ(6))
  00442   C      .
  00443   C      .                                              ID=5
  00444          .  CALL SUBDF (LCNTRL,25,1)
  00445          .  CALL AREA (1)
  00446          .  CALL INTENS (6)
  00447          .  CALL CHRSCL (0)
  00448          .  CALL LPHIT(1)
  00449          .  CALL NAME (5)
  00450          .  CALL APOS (10,300)
  00451          .  CALL TEXT ('RETURN TO CONTROL')
  00452          .  CALL LPHIT(0)
  00453          .  CALL AREA (0)
  00454          .  CALL ENDSB(ISZ(7))
  00455   C      .
  00456   C      .                                              ID=6
  00457          .  CALL SUBDF (LPAGE2,25,1)
  00458          .  CALL AREA (1)
  00459          .  CALL INTENS (6)
  00460          .  CALL CHRSCL (0)
  00461          .  CALL LPHIT(1)
  00462          .  CALL NAME (6)
  00463          .  CALL APOS (10,260)
  00464          .  CALL TEXT ('DISPLAY WELL SELECT PAGE')
  00465          .  CALL LPHIT(0)
  00466          .  CALL AREA (0)
  00467          .  CALL ENDSB(ISZ(8))
  00468   C      .
  00469   C      .                                              ID=7
  00470          .  CALL SUBDF (LPAGE1,25,1)
  00471          .  CALL AREA (1)
  00472          .  CALL INTENS (6)
  00473          .  CALL CHRSCL (0)
  00474          .  CALL LPHIT(1)
  00475          .  CALL NAME (7)
  00476          .  CALL APOS (10,220)
  00477          .  CALL TEXT ('DISPLAY DATA SELECT PAGE')
  00478          .  CALL LPHIT(0)
  00479          .  CALL AREA (0)
  00480          .  CALL ENDSB
  00481   C      .
  00482          ...FIN

         ----------------------------------------

  00483          TO SETUP-FORMAT-SPACING-LINE
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  0010
  00484          .  CALL INTENS (3)
  00485          .  IXL = ISPACE*11
  00486          .  IY1 = ISPACE/2
  00487          .  IY2 = ISPACE/4
  00488          .  CALL APOS (IX,IY)
  00489          .  CALL RVEC (0,-IY1)
  00490          .  CALL RVEC (IXL,0)
  00491          .  IYS = IY-IY1
  00492          .  IXS = IX+ISPACE
  00493          .  IK = 3
  00494          .  DO (J=1,11)
  00495          .  .  CALL APOS (IXS,IYS)
  00496          .  .  IF(IK.EQ.4) CALL RVEC (0,IY1)
  00497          .  .  IF(IK.NE.4) CALL RVEC (0,IY2)
  00498          .  .  IXS = IXS+ISPACE
  00499          .  .  IK = IK+1
  00500          .  .  IF(IK.EQ.5) IK = 1
  00501          .  ...FIN
  00502          ...FIN

         ----------------------------------------

  00503          TO SETUP-MATRIX-DISPLAY
  00504    C     .
  00505          .  CALL SUBOF (MATRIX,1345,1)
  00506          .  CALL CHRSCL (1)
  00507          .  CALL INTENS (5)
  00508          .  IY = 850
  00509          .  CALL APOS (95,IY)
  00510          .  CALL TEXT ('SELECT A DATA TYPE AND AN OUTPUT DEVICE WITH')
  00511          .  CALL TEXT (' THE LIGHT PEN:',-4)
  00512          .  CALL RVEC (815,0)
  00513          .  IY = IY-40
  00514          .  DO (J=1,2)
  00515          .  .  CALL APOS (240,IY)
  00516          .  .  CALL BOX (20,20)
  00517          .  .  CALL APOS (244,IY+4)
  00518          .  .  SELECT (J)
  00519          .  .  .  (1) CALL TEXT ('A  - INDICATES DATA AND DEVICE AVAILABILITY')
  00520          .  .  .  (2) CALL TEXT ('C  - INDICATES THE CURRENT SELECTION')
  00521          .  .  ...FIN
  00522          .  .  IY = IY-25
  00523          .  ...FIN
  00524          .  CALL CHRSCL (2)
  00525          .  CALL CHRTYP (1,1,0)
  00526          .  CALL APOS (30,210)
  00527          .  CALL TEXT ('DATA TYPES')
  00528    C     .
  00529          .  CALL APOS (50,0)
  00530          .  CALL BOX (170,630)
  00531          .  IY = 70
  00532          .  DO (J=1,8)
  00533          .  .  CALL APOS (50,IY)
  00534          .  .  CALL RVEC (170,0)
  00535          .  .  IY = IY+70
  00536          .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00011
  00537   C       .
  00538           .  CALL CHRSCL (0)
  00539           .  CALL CHRTYP (0,0,0)
  00540           .  CALL APOS (90,600)
  00541           .  DO (J=1,9)
  00542           .  .  SELECT (J)
  00543           .  .  .  (1)
  00544           .  .  .  .  CALL TEXT ('WATER LEVEL',-17)
  00545           .  .  .  .  CALL TEXT ('HISTORIES',-53)
  00546           .  .  .  ...FIN
  00547           .  .  .  (2)
  00548           .  .  .  .  CALL TEXT ('CONTAMINATION',-17)
  00549           .  .  .  .  CALL TEXT ('HISTORIES',-53)
  00550           .  .  .  ...FIN
  00551           .  .  .  (3)
  00552           .  .  .  .  CALL TEXT ('WELL STRUCTURE',-17)
  00553           .  .  .  .  CALL TEXT ('DOCUMENTATION',-53)
  00554           .  .  .  ...FIN
  00555           .  .  .  (4)
  00556           .  .  .  .  CALL TEXT ('WELL LOG',-17)
  00557           .  .  .  .  CALL TEXT ('RECORDS',-53)
  00558           .  .  .  ...FIN
  00559           .  .  .  (5)
  00560           .  .  .  .  CALL TEXT ('SOIL SIEVE',-17)
  00561           .  .  .  .  CALL TEXT ('ANALYSIS',-53)
  00562           .  .  .  ...FIN
  00563           .  .  .  (6)
  00564           .  .  .  .  CALL TEXT ('PUMP TEST',-17)
  00565           .  .  .  .  CALL TEXT ('RESULTS',-53)
  00566           .  .  .  ...FIN
  00567           .  .  .  (7)
  00568           .  .  .  .  CALL TEXT ('PHYS. PROP.',-17)
  00569           .  .  .  .  CALL TEXT ('OF SOILS',-53)
  00570           .  .  .  ...FIN
  00571           .  .  .  (8)
  00572           .  .  .  .  CALL TEXT ('CHEM. PROP.',-17)
  00573           .  .  .  .  CALL TEXT ('OF SOILS',-53)
  00574           .  .  .  ...FIN
  00575           .  .  .  (9)
  00576           .  .  .  .  CALL TEXT ('TEMPERATURE',-17)
  00577           .  .  .  .  CALL TEXT ('HISTORIES')
  00578           .  .  .  ...FIN
  00579           .  .  ...FIN
  00580           .  ...FIN
  00581   C       .
  00582           .  CALL CHRSCL (2)
  00583           .  CALL CHRTYP (1,0,0)
  00584           .  CALL APOS (473,720)
  00585           .  CALL TEXT ('OUTPUT DEVICES')
  00586           .  CALL APOS (220,0)
  00587           .  CALL BOX (800,700)
  00588           .  CALL APOS (220,630)
  00589           .  CALL RVEC (800,0)
  00590           .  IX = 380
  00591           .  DO (J=1,4)
  00592           .  .  CALL APOS (IX,640)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00012
  00593          .  .  CALL RVEC (0,60)
  00594          .  .  IX = IX+160
  00595          .  ...FIN
  00596          .  CALL CHRSCL (0)
  00597          .  CALL CHRTYP (0,0,0)
  00598          .  IX = 266
  00599          .  DO (J=1,5)
  00600          .  .  CALL APOS (IX,670)
  00601          .  .  SELECT (J)
  00602          .  .  .  (1)
  00603          .  .  .  .  CALL TEXT ('  VS60',-17)
  00604          .  .  .  .  CALL TEXT (' DISPLAY')
  00605          .  .  .  ...FIN
  00606          .  .  .  (2)
  00607          .  .  .  .  CALL TEXT ('CAL-COMP',-17)
  00608          .  .  .  .  CALL TEXT ('PLOTTER')
  00609          .  .  .  ...FIN
  00610          .  .  .  (3)
  00611          .  .  .  .  CALL TEXT ('  LINE',-17)
  00612          .  .  .  .  CALL TEXT (' PRINTER')
  00613          .  .  .  ...FIN
  00614          .  .  .  (4)
  00615          .  .  .  .  CALL TEXT ('  GOULD',-17)
  00616          .  .  .  .  CALL TEXT (' PRINTER')
  00617          .  .  .  ...FIN
  00618          .  .  .  (5)
  00619          .  .  .  .  CALL TEXT ('TEKTRONIX',-17)
  00620          .  .  .  .  CALL TEXT (' DISPLAY')
  00621          .  .  .  ...FIN
  00622          .  .  ...FIN
  00623          .  .  IX = IX+160
  00624          .  ...FIN
  00625          .  IY = 595
  00626          .  DO (J=1,9)
  00627          .  .  IX1 = 220
  00628          .  .  IX2 = 290
  00629          .  .  CALL APOS (IX1,IY)
  00630          .  .  CALL RVEC ((IX2-IX1),0)
  00631          .  .  DO (I=1,5)
  00632          .  .  .  IYB = IY-10
  00633          .  .  .  CALL APOS (IX2,IYB)
  00634          .  .  .  CALL BOX (20,20)
  00635          .  .  .  IF (J.EQ.1)
  00636          .  .  .  .  CALL APOS (IX2+10,IYB+20)
  00637          .  .  .  .  CALL RVEC (0,25)
  00638          .  .  .  ...FIN
  00639          .  .  .  CALL APOS (IX2+10,IYB)
  00640          .  .  .  WHEN (J.EQ.9) CALL RVEC (0,-25)
  00641          .  .  .  ELSE
  00642          .  .  .  .  CALL RVEC (0,-50)
  00643          .  .  .  ...FIN
  00644          .  .  .  IX1 = IX2+20
  00645          .  .  .  WHEN (I.EQ.5) IX2 = IX2+90
  00646          .  .  .  ELSE
  00647          .  .  .  .  IX2 = IX2+160
  00648          .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00013
  00649          .  .  .  CALL APOS (IX1,IY)
  00650          .  .  .  CALL RVEC ((IX2-IX1),0)
  00651          .  .  ...FIN
  00652          .  .  IY = IY-70
  00653          .  ...FIN
  00654          .  CALL ENDSB(ISZ(9))
  00655    C     .
  00656    C     .                                              ID=10
  00657          .  CALL SUBDF (NXTPG1,35,1)
  00658          .  CALL AREA (1)
  00659          .  CALL INTENS (6)
  00660          .  CALL CHRSCL (0)
  00661          .  CALL LPHIT(1)
  00662          .  CALL NAME (10)
  00663          .  CALL APOS (10,450)
  00664          .  CALL TEXT ('CONTINUE')
  00665          .  CALL LPHIT(0)
  00666          .  CALL APOS (0,440)
  00667          .  CALL BOX (75,27)
  00668          .  CALL INTENS (1)
  00669          .  CALL APOS (3,443)
  00670          .  CALL BOX (69,21)
  00671          .  CALL AREA (0)
  00672          .  CALL ENDSB(ISZ(10))
  00673    C     .
  00674    C     .                                              ID=11
  00675          .  CALL SUBDF (NRTRV,40,1)
  00676          .  CALL AREA (1)
  00677          .  CALL INTENS (6)
  00678          .  CALL CHRSCL (0)
  00679          .  CALL LPHIT(1)
  00680          .  CALL NAME (11)
  00681          .  CALL APOS (10,500)
  00682          .  CALL TEXT ('RETRIEVE',-13)
  00683          .  CALL TEXT ('DATA')
  00684          .  CALL LPHIT(0)
  00685          .  CALL APOS (0,477)
  00686          .  CALL BOX (75,40)
  00687          .  CALL INTENS (1)
  00688          .  CALL APOS (3,480)
  00689          .  CALL BOX (69,34)
  00690          .  CALL ENDSB(ISZ(11))
  00691    C     .
  00692          .  CALL SUBDF (MESS2,55,1)
  00693          .  CALL AREA (1)
  00694          .  CALL INTENS (6)
  00695          .  CALL CHRSCL (0)
  00696          .  CALL BLINK (1)
  00697          .  CALL APOS (0,500)
  00698          .  CALL TEXT ('DATA ACCESS',-15)
  00699          .  CALL TEXT ('IN PROGRESS')
  00700          .  CALL BLINK (0)
  00701          .  CALL ENDSB
  00702          ...FIN

         ----------------------------------------
```

```
00703          TO SETUP-MATRIX-SELECT-BUTTONS
00704    C     .
00705    C     .                                                ID=100 TO 204
00706          .  CALL SUBDF (MTXSEL,160,IMTX)
00707          .  CALL INTENS (4)
00708          .  CALL LPHIT (1)
00709          .  IMTX = 3
00710          .  DO (J=1,11)
00711          .  .  IF (J.LT.9 .OR. J.GT.10)
00712          .  .  .  IDT = J-1
00713          .  .  .  III = IDT
00714          .  .  .  IZ = IFLD (IDT,1,IDBIT)
00715          .  .  .  IF (IZ.EQ.1)
00716          .  .  .  .  DO (K=1,5)
00717          .  .  .  .  .  IDT = K-1
00718          .  .  .  .  .  IZ = IFLD (IDT,1,IBIT(K))
00719          .  .  .  .  .  IF (IZ.EQ.1)
00720          .  .  .  .  .  .  ID = 100 + IDT*10 + IDT
00721          .  .  .  .  .  .  CALL NAME (ID)
00722          .  .  .  .  .  .  IX = IDT*160 + 295
00723          .  .  .  .  .  .  IF (IDT .EQ. 10) III = 8
00724          .  .  .  .  .  .  IY = 500 - III*70
00725          .  .  .  .  .  .  CALL APOS (IX,IY)
00726          .  .  .  .  .  .  WHEN (IDT.EQ.IDTYP. AND .IDT.EQ.IDTYP)
00727          .  .  .  .  .  .  .  CALL TEXT ('C')
00728          .  .  .  .  .  .  ...FIN
00729          .  .  .  .  .  .  ELSE
00730          .  .  .  .  .  .  .  CALL TEXT ('A')
00731          .  .  .  .  .  .  ...FIN
00732          .  .  .  .  .  ...FIN
00733          .  .  .  .  ...FIN
00734          .  .  .  ...FIN
00735          .  .  ...FIN
00736          .  ...FIN
00737          .  CALL LPHIT (0)
00738          .  CALL ENDSB
00739          ...FIN

       ----------------------------------------

00740          TO SETUP-CONTAMINANT-NAMES-DISPLAY
00741    C     .
00742          .  CALL SUBDF (NAMCTM,560,INCTM)
00743          .  INCTM = 3
00744          .  CALL INTENS (4)
00745          .  CALL CHRSCL (1)
00746          .  IX = 200
00747          .  IY = 800
00748          .  DO (J=1,32)
00749          .  .  CALL APOS (IX,IY)
00750          .  .  CALL BOX (20,20)
00751          .  .  CALL APOS (IX+50,IY+4)
00752          .  .  SELECT (J)
00753          .  .  .  (1)  CALL TEXT ('TOTAL ALPHA')
00754          .  .  .  (2)  CALL TEXT ('TOTAL BETA')
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00015
  00755          .  .  .  (3)  CALL TEXT ('TRITIUM')
  00756          .  .  .  (4)  CALL TEXT ('NITRATE')
  00757          .  .  .  (5)  CALL TEXT ('STRONTIUM-90')
  00758          .  .  .  (6)  CALL TEXT ('CESIUM-137')
  00759          .  .  .  (7)  CALL TEXT ('COBALT-60')
  00760          .  .  .  (8)  CALL TEXT ('URANIUM-238')
  00761          .  .  .  (9)  CALL TEXT ('PLUTONIUM-239')
  00762          .  .  .  (10) CALL TEXT ('RUTHENIUM-106')
  00763          .  .  .  (11) CALL TEXT ('CHROMIUM')
  00764          .  .  .  (12) CALL TEXT ('FLUORIDE')
  00765          .  .  .  (13) CALL TEXT ('MAGNESIUM')
  00766          .  .  .  (14) CALL TEXT ('IRON')
  00767          .  .  .  (15) CALL TEXT ('PHOSPHATE')
  00768          .  .  .  (16) CALL TEXT ('CHLORIDE')
  00769          .  .  .  (17) CALL TEXT ('COPPER')
  00770          .  .  .  (18) CALL TEXT ('HARDNESS')
  00771          .  .  .  (19) CALL TEXT ('SOLIDS')
  00772          .  .  .  (20) CALL TEXT ('MANGANESE')
  00773          .  .  .  (21) CALL TEXT ('TOTAL ORGANIC CARBON')
  00774          .  .  .  (22) CALL TEXT ('PH')
  00775          .  .  .  (23) CALL TEXT ('SULFATE')
  00776          .  .  .  (24) CALL TEXT ('SODIUM')
  00777          .  .  .  (25) CALL TEXT ('CALCIUM')
  00778          .  .  .  (26) CALL TEXT ('TOTAL GAMMA')
  00779          .  .  .  (27) CALL TEXT ('BICARBONATE ION')
  00780          .  .  .  (28) CALL TEXT ('CARBONATE ION')
  00781          .  .  .  (29) CALL TEXT ('TOTAL POTASSIUM')
  00782          .  .  .  (30) CALL TEXT ('SPECIFIC CONDUCTIVITY')
  00783          .  .  .  (31) CALL TEXT ('BORON')
  00784          .  .  .  (32) CALL TEXT ('LOW ALPHA')
  00785          .  .  ...FIN
  00786          .  .  IF (J.EQ.16)
  00787          .  .  .  IX = 600
  00788          .  .  .  IY = 850
  00789          .  .  ...FIN
  00790          .  .  IY = IY-50
  00791          .  ...FIN
  00792          .  CALL ENDSR(ISZ(12))
  00793          ...FIN

         ----------------------------------------

  00794          TO SETUP-CONTAMINANT-LIGHT-BUTTONS
  00795    C     .
  00796    C     .                                          ID=901 TO 932
  00797          .  CALL SUBOF (CTMSEL,200,ICSEL)
  00798          .  ICSEL = 3
  00799          .  CALL INTENS (4)
  00800          .  CALL CHRSCL (1)
  00801          .  CALL LPHIT (1)
  00802          .  DO (J=1,32)
  00803          .  .  WHEN (J.LE.16)
  00804          .  .  .  IZ = IFLD((J-1),1,ICBIT(1))
  00805          .  .  ...FIN
  00806          .  .  ELSE
  00807          .  .  .  IZ = IFLD((J-17),1,ICBIT(2))
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00016
  00808          .  .  ...FIN
  00809          .  .  IF (IZ.EQ.1)
  00810          .  .  .  ID = 900+.1
  00811          .  .  .  CALL NAME (ID)
  00812          .  .  .  IX = 204 + (J/17)*400
  00813          .  .  .  IY = 854 - J*50 + (J/17)*800
  00814          .  .  .  CALL APOS (IX,IY)
  00815          .  .  .  WHEN (J.EQ.ICTYP) CALL TEXT ('C')
  00816          .  .  .  ELSE
  00817          .  .  .  .  CALL TEXT ('A')
  00818          .  .  .  ...FIN
  00819          .  .  ...FIN
  00820          .  ...FIN
  00821          .  CALL LPHIT (0)
  00822          .  CALL ENDSR
  00823          .  CALL STRTSB (CTMSEL)
  00824          ...FIN
  00825          END

          ----------------------------------------

                 PROCEDURE CROSS-REFERENCE TABLE

          00284 DISPLAY-WELL-LOCATION-MAP
                 00165  00196

          00293 SEE-IF-WELL-EXISTS
                 00117

          00794 SETUP-CONTAMINANT-LIGHT-BUTTONS
                 00095  00210  00275

          00740 SETUP-CONTAMINANT-NAMES-DISPLAY
                 00076

          00324 SETUP-DISPLAY-FILES
                 00074

          00483 SETUP-FORMAT-SPACING-LINE
                 00384  00413

          00503 SETUP-MATRIX-DISPLAY
                 00075

          00703 SETUP-MATRIX-SELECT-BUTTONS
                 00090  00175  00264

          00301 SETUP-WELL-NAME-DISPLAY
                 00088  00115

          00310 SETUP-WELL-PARAMETERS-DISPLAY
                 00089  00119


          (FLECS VERSION 22.46)


(FLECS VERSION 22.46)     14-SEP-79   07:58:22  PAGE  00017
          ----------------------------------------

  00826          SUBROUTINE TRAP
  00827          STOP 'UNEXPECTED INTERRUPT RECIEVED'
  00828          END

          (FLECS VERSION 22.46)

          ----------------------------------------
```

HYDSCP.FLX

HYDROGRAPH (SCOPE)

```
(FLECS VERSION 22.46)     14-SEP-79   07:59:28  PAGE  00001

      ----------------------------------------

00001   C ******************** (351,101]HYDSCP.FLX ********************
00002   C
00003   C
00004   C  PROGRAM ID: SRV-55-WEL-0-0 ABREVIATED WELL DESIG, VERSION
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL HYDROGRAPH DATA
00008   C     FROM THE CIRMIS DATA BANK AND DISPLAY A PLOT OF THE DATA.
00009   C
00010   C  DATA FILES:
00011   C       NAME                        LUN          TYPE          ACCESS
00012   C       FILE0(D80)---FRMTHYD         3           RAN            R
00013   C       FILE0(DK1)---WELSPEC         4           RAN            R/W
00014   C
00015   C
00016   C
00017   C
00018   C  BATTELLE MEMORIAL INSTITUTE
00019   C  PACIFIC NORTHWEST LABORATORIES
00020   C  WATER & LAND RESOURCES DEPT.
00021   C
00022   C  AUTHOR(S):   DR FRIEDRICHS
00023   C               DW DAMSCHEN
00024   C               VL COBURN
00025   C
00026   C  DATE:  INITIAL VERSION  FEBRUARY 4, 1978
00027   C         CURRENT VERSION  MARCH 5, 1979
00028   C
00029   C
00030         BYTE WLDES,WNAME,KEYBUF(80)
00031         INTEGER BKGRND,BKGLIN, WORKSP(35), PASSWD(4), STATUS(2), UIC(2)
00032         LOGICAL IN,LASTIN
00033   C
00034         DIMENSION H(512),T(512),INODE(280),ISZ(9),MSGBUF(20)
00035   C
00036         COMMON /DSP/MAIN(100),KEYOUT(100),BKGRND(100),BKGLIN(700),
00037        1         LBUT1(300),MESSY1(25),MESSY2(25),MESSX1(30),MESSX2(30),
00038        2         IGRAPH(5000),MESS2(35)
00039   C
00040         COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IDBIT,ICBIT(2),
00041        1            IDM(236),NRETRN,MODE,WNAME(12),IOTYP,IOTYP,
00042        2            ICTYP,NVAR,MINX,MINY,MAXX,MAXY,IDM3(7)
00043   C
00044         EQUIVALENCE (WLDES(1),INODE(1)),(MSGBUF(1),WNAME(1))
00045   C
00046   C
00047         CALL MAINUP (MAIN,100,0,1,0,2)
00048         CALL CLKUPP
00049   C
00050         CALL OPFILE (3,'FRMTHYD',4000,,IPNT1)
00051   C
00052         SETUP-LIGHT-BUTTONS
00053   C
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:59:28  PAGE  00002
  00054          IHG = 0
  00055          IRL = 0
  00056          IKB = 1
  00057    C
  00058          RETRIEVE-HYDROGRAPH-DATA
  00059          CALCULATE-SCALE-FACTORS
  00060          DRAW-THE-BACKGROUND-CHART
  00061          DISPLAY-X-AND-Y-LINES-FOR-CHART
  00062          PLOT-DATA-POINTS-ON-GRAPH
  00063    C     WRITE (5,551) ISZ
  00064    551   FORMAT (' ISZ=',914)
  00065    C
  00066    C*************************************************************
  00067    C
  00068    C     START GRAPHIC CONTROL
  00069    C
  00070    C
  00071    C*************************************************************
  00072    C
  00073          IKB = 0
  00074    10    CALL GRATTN (1,ITYP,'LP')
  00075          IF(ITYP.NE.'LP') GO TO 10
  00076    C *** LIGHT PEN INTERRUPTS ***
  00077          CALL LTPEN (ID,ITIP)
  00078          IF(ITIP.EQ.1)
  00079          .  CONDITIONAL
  00080          .  .  (ID.EQ.1)
  00081          .  .  .  CALL STRTSB (MESSY1)
  00082          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00083          .  .  .  DECODE (NCHAR,219,KEYBUF) MINY
  00084    219   .  .  .  FORMAT (I4)
  00085          .  .  .  CALL STOPSB (MESSY1)
  00086          .  .  .  CALL STRTSB (MESSY2)
  00087          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00088          .  .  .  DECODE (NCHAR,219,KEYBUF) MAXY
  00089          .  .  .  CALL STOPSB (MESSY2)
  00090          .  .  .  YDIF = (MAXY-MINY)
  00091          .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00092          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00093          .  .  ...FIN
  00094          .  .  (ID.EQ.2)
  00095          .  .  .  CALL STRTSB (MESSX1)
  00096          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00097          .  .  .  DECODE (NCHAR,219,KEYBUF) MINX
  00098          .  .  .  CALL STOPSB (MESSX1)
  00099          .  .  .  CALL STRTSB (MESSX2)
  00100          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00101          .  .  .  DECODE (NCHAR,219,KEYBUF) MAXX
  00102          .  .  .  CALL STOPSB (MESSX2)
  00103          .  .  .  XDIF = (MAXX-MINX)+1
  00104          .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00105          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00106          .  .  ...FIN
  00107          .  .  (ID.EQ.3)
  00108          .  .  .  MINX = MINXOR
  00109          .  .  .  MAXX = MAXXOR
```

```
'FLECS VERSION 22.46)      14-SEP-79   07:59:28  PAGE  00003
  00110          . . .  MINY = MINYUR
  00111          . . .  MAXY = MAXYUR
  00112          . . .  YDIF = (MAXY-MINY)
  00113          . . .  XDIF = (MAXX-MINX)+1
  00114          . . .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00115          . . .  PLOT-DATA-POINTS-ON-GRAPH
  00116          . . ...FIN
  00117          . .  (ID.EQ.4)
  00118          . . .  IOTYP = 1
  00119          . . .  ICON = 0
  00120          . . .  PASSWD(1) = 351
  00121          . . .  CALL NTINIT (STATUS,35,WORKSP)
  00122          . . .  CALL NTCONW (6,STATUS,ICON,'WLR45_','MNTR45',,UIC,1,PASSWD,TRAP)
  00123          . . .  CALL NTSNDW (6,STATUS,40,MSGBUF)
  00124          . . .  CONDITIONAL
  00125          . . .  .  (STATUS(1).LT.1)
  00126          . . .  .  .  WRITE(5,535) STATUS(1)
  00127   535    . . .  .  .  FORMAT(' TARGET IS NOT RECEPTIVE, STATUS =',I5)
  00128          . . .  .  .  CALL NTWAIT (6,STATUS)
  00129          . . .  .  ...FIN
  00130          . . .  .  (STATUS(1).EQ.1)
  00131          . . .  .  .  CALL NTDISW (6,STATUS,,)
  00132          . . .  .  ...FIN
  00133          . . .  .  (STATUS(1).GT.1)
  00134          . . .  .  .  WRITE(5,536) STATUS(1)
  00135   536    . . .  .  .  FORMAT(' ERROR DURING COMMUNICATION--CODE=',I5)
  00136          . . .  .  ...FIN
  00137          . . .  ...FIN
  00138          . . .  CALL NTDISW (6,STATUS)
  00139          . .  ...FIN
  00140          . .  (ID.GE.20. AND .ID.LE.22)
  00141          . . .  NOCORD = 0
  00142          . . .  SELECT (ID)
  00143          . . .  .  (20)
  00144          . . .  .  .  NRETRN = 0
  00145          . . .  .  .  TASK = RAD50 ('CIRMIS')
  00146          . . .  .  ...FIN
  00147          . . .  .  (21)
  00148          . . .  .  .  NRETRN = 1
  00149          . . .  .  .  TASK = RAD50 ('WELSEL')
  00150          . . .  .  ...FIN
  00151          . . .  .  (22)
  00152          . . .  .  .  WHEN (XC.NE.0. AND .YC.NE.0)
  00153          . . .  .  .  .  NRETRN = 20
  00154          . . .  .  .  .  TASK = RAD50('MAPLOC')
  00155          . . .  .  .  ...FIN
  00156          . . .  .  .  ELSE
  00157          . . .  .  .  .  CALL STRTSB (MES32)
  00158          . . .  .  .  .  CALL WAIT (2,2,MMM)
  00159          . . .  .  .  .  CALL STOPSB (MES32)
  00160          . . .  .  .  .  NOCORD = 1
  00161          . . .  .  .  ...FIN
  00162          . . .  .  ...FIN
  00163          . . .  ...FIN
  00164          . . .  IF(NOCORD.EW.0)
  00165          . . .  .  CALL DPW (IPNT0,0.,,INODE,200.)
```

```
(FLECS VERSION 22.06)      14-SEP-79   07:59:28  PAGE  00004
  00166          .  .  .  .  CALL REQUES (TASK,,IOS)
  00167          .  .  .  .  III = 0
  00168          .  .  .  .  IF(III.EQ.0) CALL EXIT
  00169          .  .  .  ...FIN
  00170          .  .  ...FIN
  00171          .  ...FIN
  00172          ...FIN
  00173          GO TO 10

         ----------------------------------------

  00174          TO RETRIEVE-HYDROGRAPH-DATA
  00175          .  CALL OPFILE (4,'WELSPEC',2,,IPNT0)
  00176          .  CALL OPR (IPNT0,0,,INODE,280,)
  00177    C     .
  00178    C  IOKIN--DISK NUMBER
  00179    C     .
  00180          .  IOKIN = INODE(30)
  00181          .  NPTSIN = INODE(31)
  00182          .  IOKEX  = INODE(32)
  00183          .  NPTSEX = INODE(33)
  00184          .  INALOC = INODE(34)
  00185    C     .
  00186    C  READ THE DATA IN THE INITIAL ALLOCATION
  00187          .  HADSI = IOKIN
  00188          .  WRDI  = NPTSIN*2.
  00189          .  TADSI = HADSI+INALOC
  00190          .  CALL OPR (IPNT1,HADSI,H,WRDI)
  00191          .  CALL OPR (IPNT1,TADSI,T,WRDI)
  00192    C     .
  00193    C  READ THE EXTENDED ALLOCATION (IF ANY)
  00194    C     .
  00195          .  IF (NPTSEX.NE.0)
  00196          .  .  HADSE = IOKEX
  00197          .  .  WRDE  = NPTSEX*2.
  00198    C     .  .
  00199    C  FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
  00200    C  HEIGHT AND TIME
  00201    C     .  .
  00202          .  .  TADSE = HADSE*2.
  00203          .  .  NDX   = NPTSIN+1
  00204          .  .  CALL OPR (IPNT1,HADSE,H(NDX),WRDE)
  00205          .  .  CALL OPR (IPNT1,TADSE,T(NDX),WRDE)
  00206          .  ...FIN
  00207          .  NPTS = NPTSIN + NPTSEX
  00208          ...FIN

         ----------------------------------------

  00209          TO CALCULATE-SCALE-FACTORS
  00210          .  IF (NVAR.EQ.0)
  00211          .  .  TMAX=0.0
  00212          .  .  TMIN=999999.
  00213          .  .  HMAX=0.0
  00214          .  .  HMIN=1.0E+20
  00215          .  .  DO (I=1,NPTS)
```

```
(FLECS VERSION 22.46)      14-SEP-79   07:59:28  PAGE  00005
  00216          .  .  .  IF(T(I).GT.TMAX) TMAX=T(I)
  00217          .  .  .  IF(T(I).LT.TMIN) TMIN=T(I)
  00218          .  .  .  IF(H(I).GT.0.)
  00219          .  .  .  .  IF(H(I).GT.HMAX) HMAX=H(I)
  00220          .  .  .  .  IF(H(I).LT.HMIN) HMIN=H(I)
  00221          .  .  .  ...FIN
  00222          .  .  ...FIN
  00223          .  .  MINX=IFIX(TMIN/10000.)-1
  00224          .  .  MAXX=IFIX(TMAX/10000.)+1
  00225          .  .  MINY = IFIX(HMIN/10.)*10-10
  00226          .  .  MAXY = IFIX(HMAX/10.)*10+10
  00227          .  ...FIN
  00228          .  NVAR = 4
  00229          .  MINXOR = MINX
  00230          .  MAXXOR = MAXX
  00231          .  MINYOR = MINY
  00232          .  MAXYOR = MAXY
  00233          .  XDIF = (MAXX-MINX)+1
  00234          .  YDIF = (MAXY-MINY)
  00235          .  INCY = 10
  00236          .  IF (YDIF.LE.100.) INCY=5
  00237          .  IF (YDIF.LE.40.)  INCY=2
  00238          .  DSPX1 = 100.
  00239          .  DSPX2 = DSPX1 + 900.
  00240          .  DSPY1 = 100.
  00241          .  DSPY2 = DSPY1 + 800.
  00242          ...FIN
  00243     C

          ----------------------------------------

  00244          TO DRAW-THE-BACKGROUND-CHART
  00245          .  CALL SUBDF (BKGRND,100,IBG)
  00246          .  IBG = 2
  00247          .  CALL INTENS (5)
  00248          .  CALL APOS (IFIX(DSPX1),IFIX(DSPY1))
  00249          .  CALL BOX (IFIX(DSPX2-DSPX1),IFIX(DSPY2-DSPY1))
  00250          .  CALL CHRSCL (1)
  00251          .  CALL CHRTYP (0,1,0)
  00252          .  CALL APOS (30,525)
  00253          .  CALL TEXT ('WATER ELEVATION  (FT-MSL)  ')
  00254          .  CALL CHRSCL (1)
  00255     1750 .  CALL APOS (450,60)
  00256          .  CALL CHRTYP (0,0,0)
  00257          .  CALL TEXT ('CALENDAR YEAR')
  00258     C    .
  00259          .  CALL CHRSCL (1)
  00260          .  CALL CHRTYP (0,0,0)
  00261          .  CALL APOS (415,950)
  00262          .  CALL TEXT ('WATER LEVEL HISTORY')
  00263          .  CALL CHRSCL (0)
  00264          .  CALL APOS (400,925)
  00265          .  CALL TEXT ('WELL DESIGNATION -- ')
  00266          .  CALL TEXT (WLDES,0,12)
  00267          .  CALL APOS (400,915)
  00268          .  CALL TEXT ('CASING ELEVATION -- ')
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:59:28  PAGE  0006
  00269          .  CALL FNMBR (CASEL,'(F6.2)')
  00270          .  CALL ENDSB(ISZ(1))
  00271          ...FIN

         ----------------------------------------

  00272          TO DISPLAY-X-AND-Y-LINES-FOR-CHART
  00273    C     .
  00274          .  CALL SUBDF (BKGLIN,700,IBL)
  00275          .  IBL = 2
  00276          .  CALL CHRSCL (0)
  00277          .  CALL INTENS (4)
  00278    C----DRAW Y LINES FOR CHART
  00279          .  DSYINC = (DSPY2-DSPY1)/YDIF
  00280          .  DO (JD = MINY,MAXY,INCY)
  00281          .  .  IYC = DSPY1 + FLOAT(JD-MINY)*DSYINC
  00282          .  .  CALL APOS (IFIX(DSPX1-30),IYC)
  00283          .  .  CALL INMBR (JD,'(I4)')
  00284          .  .  IF (JD.GT.MINY. AND .JD.LT.MAXY)
  00285          .  .  .  CALL APOS (IFIX(DSPX1),IYC)
  00286          .  .  .  CALL VEC  (IFIX(DSPX2),IYC)
  00287          .  .  ...FIN
  00288          .  ...FIN
  00289    C----DRAW X LINES FOR CHART
  00290          .  JDD = 1
  00291          .  DSXINC = (DSPX2-DSPX1)/XDIF
  00292          .  XADJ = DSXINC/2. + 7.
  00293          .  DO (JD = MINX,MAXX+1)
  00294          .  .  X = DSPX1 + FLOAT(JD-MINX+1)*DSXINC
  00295          .  .  IF(JD.LE.MAXX)
  00296          .  .  .  CALL APOS (IFIX(X-XADJ),IFIX(DSPY1-15))
  00297          .  .  .  CALL INMBR (JD,'(I2)')
  00298          .  .  ...FIN
  00299          .  .  IF(JD.LE.MAXX)
  00300          .  .  .  CALL APOS (IFIX(X),IFIX(DSPY1))
  00301          .  .  .  SELECT (JDD)
  00302          .  .  .  .  (1)
  00303          .  .  .  .  .  CALL RVEC (0,10)
  00304          .  .  .  .  .  JDD = 2
  00305          .  .  .  .  ...FIN
  00306          .  .  .  .  (2)
  00307          .  .  .  .  .  CALL RVEC (0,IFIX(DSPY2-DSPY1))
  00308          .  .  .  .  .  JDD = 1
  00309          .  .  .  .  ...FIN
  00310          .  .  .  ...FIN
  00311          .  .  ...FIN
  00312          .  ...FIN
  00313          .  CALL ENDSB(ISZ(2))
  00314          ...FIN

         ----------------------------------------

  00315          TO PLOT-DATA-POINTS-ON-GRAPH
  00316          .  CALL SUBDF (IGRAPH,5000,I38)
  00317          .  I38 = 2
  00318          .  CALL INTENS (6)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07:59:28  PAGE  00007
  00319          .  XSCALE = (DSPX2-DSPX1)/(XDIF*365.)
  00320          .  LASTIN = .FALSE.
  00321          .  DO (JK=1,NPTS)
  00322          .  .  IYR   = T(JK)/10000.
  00323          .  .  MO    = T(JK)/100.-FLOAT(IYR)*100.
  00324          .  .  IDA   = T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
  00325    38    .  .  DAYS = (FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
  00326          .  .  IX    = DAYS*XSCALE+DSPX1
  00327          .  .  X     = IX
  00328          .  .  IY = DSPY1 + (H(JK)-FLOAT(MINY))*DSYINC
  00329          .  .  Y   = IY
  00330    C  CHECK FOR POINT WITHIN REGION
  00331          .  .  IN = .FALSE.
  00332          .  .  IF(X.GE.DSPX1. AND .X.LE.DSPX2)
  00333          .  .  .  IF(Y.GE.DSPY1. AND .Y.LE.DSPY2)
  00334          .  .  .  .  WHEN (LASTIN) CALL VEC (IX,IY)
  00335          .  .  .  .  ELSE
  00336          .  .  .  .  .  CALL APOS (IX,IY)
  00337          .  .  .  .  ...FIN
  00338          .  .  .  .  LASTIN = .TRUE.
  00339          .  .  .  .  IN = .TRUE.
  00340          .  .  .  ...FIN
  00341          .  .  ...FIN
  00342          .  .  IF (.NOT.IN) LASTIN = .FALSE.
  00343          .  ...FIN
  00344          .  CALL ENDSR(ISZ(4))
  00345          .  CALL STRTSR (IGRAPH)
  00346          ...FIN

         ----------------------------------------

  00347          TO SETUP-LIGHT-BUTTONS
  00348    C     .
  00349          .  CALL SUBDF (LAUT1,300,0)
  00350          .  CALL AREA (1)
  00351          .  CALL CHRSCL (0)
  00352          .  CALL INTENS (4)
  00353          .  CALL APOS (0,700)
  00354          .  CALL TEXT ('  LIGHT PEN OPTIONS')
  00355          .  CALL INTENS (7)
  00356          .  CALL APOS (0,685)
  00357          .  CALL BOX (150,40)
  00358          .  CALL APOS (5,690)
  00359          .  CALL BOX (140,30)
  00360          .  CALL APOS (30,625)
  00361          .  CALL TEXT ('SCALE CHANGE',-3)
  00362          .  CALL RVEC (A4,0)
  00363          .  CALL MENU (590,30,1,'ELEVATION SCALE','TIME SCALE',
  00364         1.  'ORIGINAL SCALE')
  00365          .  CALL AREA (1)
  00366          .  CALL APOS (30,450)
  00367          .  CALL TEXT ('OUTPUT DEVICES',-3)
  00368          .  CALL RVEC (90,0)
  00369          .  CALL MENU (415,30,4,'CAL-COMP',
  00370         1.  'GOULD','TEKTRONIX')
  00371          .  CALL AREA (1)
```

```
(FLECS VERSION 22.46)     14-SEP-79   07159128  PAGE  00008
  00372           .  CALL APOS (30,150)
  00373           .  CALL TEXT ('PROGRAM CONTROLS',-3)
  00374           .  CALL RVEC (110,0)
  00375           .  CALL MENU (115,34,20,'RETURN TO CIRMIS CONTROL',
  00376          1.  'RETURN TO WELL SELECT','DISPLAY MAP LOCATION')
  00377           .  CALL AREA (0)
  00378           .  CALL ENDSB(ISZ(5))
  00379    C      .
  00380    C *** OUTPUT MESSAGES
  00381           .  CALL SUBDF (MESSY1,25,1)
  00382           .  CALL CHRSCL (0)
  00383           .  CALL INTENS (7)
  00384           .  CALL BLINK (1)
  00385           .  CALL APOS (0,30)
  00386           .  CALL TEXT ('ENTER MINIMUM Y ELEVATION (I4)')
  00387           .  CALL BLINK (0)
  00388           .  CALL ENDSB(ISZ(6))
  00389    C      .
  00390           .  CALL SUBDF (MESSY2,25,1)
  00391           .  CALL CHRSCL (0)
  00392           .  CALL INTENS (7)
  00393           .  CALL BLINK (1)
  00394           .  CALL APOS (0,50)
  00395           .  CALL TEXT ('ENTER MAXIMUM Y ELEVATION (I4)')
  00396           .  CALL BLINK (0)
  00397           .  CALL ENDSB (ISZ(7))
  00398    C      .
  00399           .  CALL SUBDF (MESSX1,30,1)
  00400           .  CALL CHRSCL (0)
  00401           .  CALL INTENS (7)
  00402           .  CALL BLINK (1)
  00403           .  CALL APOS (0,30)
  00404           .  CALL TEXT ('ENTER MINIMUM TIME IN YEARS (I4)')
  00405           .  CALL BLINK (0)
  00406           .  CALL ENDSB(ISZ(8))
  00407    C      .
  00408           .  CALL SUBDF (MESSX2,30,1)
  00409           .  CALL CHRSCL (0)
  00410           .  CALL INTENS (7)
  00411           .  CALL BLINK (1)
  00412           .  CALL APOS (0,50)
  00413           .  CALL TEXT ('ENTER MAXIMUM TIME IN YEARS (I4)')
  00414           .  CALL BLINK (0)
  00415           .  CALL ENDSB(ISZ(9))
  00416    C      .
  00417           .  CALL SUBDF (MESS2,35,1)
  00418           .  CALL CHRSCL (0)
  00419           .  CALL AREA (1)
  00420           .  CALL BLINK (1)
  00421           .  CALL APOS (0,200)
  00422           .  CALL TEXT ('COORDINATES NOT AVAILABLE FOR THIS WELL')
  00423           .  CALL BLINK (0)
  00424           .  CALL AREA (0)
  00425           .  CALL ENDSB
  00426           ...FIN
  00427          END
```

PROCEDURE CROSS-REFERENCE TABLE

00209 CALCULATE-SCALE-FACTORS
00059

00272 DISPLAY-X-AND-Y-LINES-FOR-CHART
00061 00091 00104 00114

00244 DRAW-THE-BACKGROUND-CHART
00060

00315 PLOT-DATA-POINTS-ON-GRAPH
00062 00092 00105 00115

00174 RETRIEVE-HYDROGRAPH-DATA
00058

00347 SETUP-LIGHT-BUTTONS
00057

(FLECS VERSION 22.46)

```
00428          SUBROUTINE GRECHO(KEYBUF,NCHAR,IX,IY,ICS,IKB)
00429    C
00430          BYTE KEYBUF(80)
00431          INTEGER DEL,CR
00432          COMMON/DSP/ MAIN(100),KEYOUT(100)
00433    C
00434          DEL = "177
00435          CR = "15
00436          CALL SUBDF(KEYOUT,100,IKB)
00437          IKB = 2
00438          CALL INTENS (4)
00439          CALL CHRSCL (ICS)
00440          CALL APOS (IX,IY)
00441          CALL TEXT ('>SE')
00442          CALL ENDSB
00443    C
00444          NCHAR = 0
00445    C
00446    10    CALL GRATTN (1,ITYP,'KB')
00447          IF (ITYP.NE.'KB') GO TO 10
00448          CALL KEYBRD (KEY)
00449          IF (KEY.EQ.CR) GO TO 30
00450          IF (NCHAR.EQ.80) GO TO 30
00451          IF (NCHAR.LT.0) GO TO 10
00452          NCHAR = NCHAR+1
00453          KEYBUF(NCHAR) = KEY
00454          IF (KEY.EQ.DEL)
00455         .  NCHAR = NCHAR-2
```

```
(FLECS VERSION 22,46)     14-SEP-79   07:59:28  PAGE  00010
  00456           ...FIN
  00457           CALL SUBUF(KEYOUT,100,IKB)
  00458           CALL INTENS (4)
  00459           CALL CHRSCL (ICS)
  00460           CALL APOS(IX,IY)
  00461           CALL TEXT (KEYBUF,0,NCHAR)
  00462           CALL TEXT ('>SE')
  00463           CALL ENDSB
  00464           GO TO 10
  00465    30     RETURN
  00466           END

          (FLECS VERSION 22,46)

          ----------------------------------------
```

CTMSCP.FLX

CONTAMINATION (SCOPE)

```
(FLECS VERSION 22.46)     14-SEP-79   08100110  PAGE  00001

      ----------------------------------------

00001   C ******************** [351,101]CTMSCP.FLX ********************
00002   C
00003   C
00004   C  PROGRAM ID: SRV-55-WEL-1-0 ABREVIATED WELL DESIG. VERSION
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL CONTAMINATE DATA
00008   C     FROM THE CIRMIS DATA BANK AND DISPLAY A PLOT OF THE DATA.
00009   C
00010   C  DATA FILES:
00011   C        NAME                        LUN          TYPE          ACCESS
00012   C        ----                        ---          ----          ------
00013   C
00014   C        VS60---DISPLAY               1           ---            R/W
00015   C        VS60---KEYBOARD              2           ---             R
00016   C        FILEQ(DB0)---WELLHDR         3           RAN             R
00017   C        FILEQ(DB0)---CTMHDR          3           RAN             R
00018   C        FILEQ(DK1)---WELSPEC         4           RAN            R/W
00019   C        NT------DECNET               6           ---            R/W
00020   C
00021   C
00022   C
00023   C  BATTELLE MEMORIAL INSTITUTE
00024   C  PACIFIC NORTHWEST LABORATORIES
00025   C  WATER & LAND RESOURCES DEPT.
00026   C
00027   C  AUTHOR(S):   DR FRIEDRICHS
00028   C               DW DAMSCHEN
00029   C               VL COBURN
00030   C
00031   C  DATE:  INITIAL VERSION  JANUARY 1978
00032   C         CURRENT VERSION   AUGUST 1979
00033   C
00034   C
00035         BYTE WLDES,WNAME,KEYBUF(80)
00036         INTEGER BKGRND,BKGLIN, WORKSP(35), PASSWD(4), STATUS(2), UIC(2)
00037         INTEGER*4 CTIADR(36), CTXADR(36)
00038         LOGICAL JOIN
00039   C
00040         DIMENSION H(400),T(400),INODE(280),IXDEL(50),IYDEL(50)
00041         DIMENSION IDEL(50),IH(800),IT(800),ISZ(10),MSGBUF(20)
00042   C
00043         COMMON /DSP/MAIN(100),KEYOUT(100),BKGRND(100),BKGLIN(900),
00044        1         LBUT1(350),MESSY1(30),MESSY2(30),MESSX1(30),MESSX2(30),
00045        2         IGRAPH(4000),MESS2(40)
00046   C
00047         COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IDBIT,ICBIT(2),
00048        1            IDM(236),NRETRN,MODE,WNAME(12),IOTYP,IDTYP,
00049        2            ICTYP,NVAR,MINX,MINY,MAXX,MAXY,MAPTYP,IDM3(6)
00050   C
00051         EXTERNAL TRAP
00052         EQUIVALENCE (WLDES(1),INODE(1)), (MSGBUF(1),WNAME(1))
00053         EQUIVALENCE (CTIADR(1),IH(1)), (CTXADR(1),IH(73))
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00002
  00054   C
  00055   C
  00056         CALL MAINOF (MAIN,100,0,1,0,2)
  00057         CALL CLKOFF
  00058   C
  00059         CALL DPFILE (3,'FRMTCTM',37000,,IPNT1)
  00060         CALL DPFILE (3,'CTMHDR',6000,,ICHDR)
  00061   C
  00062         SETUP-LIGHT-BUTTONS
  00063   C     WRITE (5,551) ISZ
  00064   C
  00065         JOIN = .FALSE.
  00066         IBG = 0
  00067         IBL = 0
  00068         IGR = 0
  00069         I00 = 0
  00070   C
  00071         RETRIEVE-CONTAMINATION-DATA
  00072         CALCULATE-SCALE-FACTORS
  00073         DRAW-THE-BACKGROUND-CHART
  00074   C     WRITE (5,551) ISZ
  00075         DISPLAY-X-AND-Y-LINES-FOR-CHART
  00076   C     WRITE (5,551) ISZ
  00077         PLOT-DATA-POINTS-ON-GRAPH
  00078   C     WRITE (5,551) ISZ
  00079   551   FORMAT (' ISZ=',10I4)
  00080   C
  00081   C*****************************************************************
  00082   C
  00083   C     START GRAPHIC CONTROL
  00084   C
  00085   C
  00086   C*****************************************************************
  00087   C
  00088         IKB = 0
  00089   10    CALL GRATIN (1,ITYP,'LP')
  00090         IF(ITYP.NE.'LP') GO TO 10
  00091   C *** LIGHT PEN INTERRUPTS ***
  00092         CALL LTPEN (ID,ITIP)
  00093         IF(ITIP.EQ.1)
  00094         .  CONDITIONAL
  00095         .  .  (ID.EQ.1)
  00096         .  .  .  CALL STRTSB (MESSY1)
  00097         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00098         .  .  .  DECODE (NCHAR,219,KEYBUF) MINY
  00099   219   .  .  .  FORMAT (I4)
  00100         .  .  .  CALL STOPSB (MESSY1)
  00101         .  .  .  CALL STRTSB (MESSY2)
  00102         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00103         .  .  .  DECODE (NCHAR,219,KEYBUF) MAXY
  00104         .  .  .  CALL STOPSB (MESSY2)
  00105         .  .  .  YDIF = (MAXY-MINY)
  00106         .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00107         .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00108         .  .  ...FIN
  00109         .  .  (ID.EQ.2)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00003
  00110          .  .  .  CALL STRTSB (MESSX1)
  00111          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00112          .  .  .  DECODE (NCHAR,219,KEYBUF) MINX
  00113          .  .  .  CALL STOPSB (MESSX1)
  00114          .  .  .  CALL STRTSB (MESSX2)
  00115          .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00116          .  .  .  DECODE (NCHAR,219,KEYBUF) MAXX
  00117          .  .  .  CALL STOPSB (MESSX2)
  00118          .  .  .  XDIF = (MAXX-MINX)+1
  00119          .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00120          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00121          .  .  ...FIN
  00122          .  .  (ID.EQ.3)
  00123          .  .  .  MINX = MINXOR
  00124          .  .  .  MAXX = MAXXOR
  00125          .  .  .  MINY = MINYOR
  00126          .  .  .  MAXY = MAXYOR
  00127          .  .  .  YDIF = (MAXY-MINY)
  00128          .  .  .  XDIF = (MAXX-MINX)+1
  00129          .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00130          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00131          .  .  ...FIN
  00132          .  .  (ID.GE.4.AND.ID.LE.6)
  00133          .  .  .  NVAR = 5
  00134          .  .  .  ICON = 0
  00135          .  .  .  PASSWD(1) = 351
  00136          .  .  .  SELECT (ID)
  00137          .  .  .  .  (4)
  00138          .  .  .  .  .  IOTYP = 1
  00139          .  .  .  .  .  MAPTYP = 0
  00140          .  .  .  .  ...FIN
  00141          .  .  .  .  (5)
  00142          .  .  .  .  .  IOTYP = 1
  00143          .  .  .  .  .  MAPTYP = 1
  00144          .  .  .  .  ...FIN
  00145          .  .  .  .  (6)
  00146          .  .  .  .  .  IOTYP = 3
  00147          .  .  .  .  ...FIN
  00148          .  .  .  ...FIN
  00149          .  .  .  CALL NTINIT (STATUS,35,WORKSP)
  00150          .  .  .  CALL NTCONW (6,STATUS,ICON,'WLR45 ','MNTR45',,UIC,1,PASSWD,TRAP)
  00151          .  .  .  CALL NTSNDW (6,STATUS,40,MSGBUF)
  00152          .  .  .  CONDITIONAL
  00153          .  .  .  .  (STATUS(1).LT.1)
  00154          .  .  .  .  .  WRITE(5,535) STATUS(1)
  00155   535    .  .  .  .  .  FORMAT(' TARGET IS NOT RECEPTIVE, STATUS =',I5)
  00156          .  .  .  .  .  CALL NTWAIT (6,STATUS)
  00157          .  .  .  .  ...FIN
  00158          .  .  .  .  (STATUS(1).EQ.1)
  00159          .  .  .  .  .  CALL NTDISW (6,STATUS,,)
  00160          .  .  .  .  ...FIN
  00161          .  .  .  .  (STATUS(1).GT.1)
  00162          .  .  .  .  .  WRITE(5,536) STATUS(1)
  00163   536    .  .  .  .  .  FORMAT(' ERROR DURING COMMUNICATION--CODE=',I5)
  00164          .  .  .  .  ...FIN
  00165          .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08100110  PAGE  00004
  00166          .  .  .  CALL NTDISW (6,STATUS)
  00167          .  .  ...FIN
  00168          .  .  (ID.EQ.7)
  00169          .  .  .  JOIN = .TRUE.
  00170          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00171          .  .  ...FIN
  00172          .  .  (ID.EQ.8)
  00173          .  .  .  JOIN = .FALSE.
  00174          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00175          .  .  ...FIN
  00176          .  .  (ID.GE.20. AND .ID.LE.22)
  00177          .  .  .  NOCORD = 0
  00178          .  .  .  SELECT (ID)
  00179          .  .  .  .  (20)
  00180          .  .  .  .  .  NRETRN = 0
  00181          .  .  .  .  .  TASK = RAD50 ('CIRMIS')
  00182          .  .  .  .  ...FIN
  00183          .  .  .  .  (21)
  00184          .  .  .  .  .  NRETRN = 2
  00185          .  .  .  .  .  IOTYP = 0
  00186          .  .  .  .  .  TASK = RAD50 ('WELSEL')
  00187          .  .  .  .  ...FIN
  00188          .  .  .  .  (22)
  00189          .  .  .  .  .  WHEN (XC.NE.0.. AND .YC.NE.0.)
  00190          .  .  .  .  .  .  NRETRN = 21
  00191          .  .  .  .  .  .  TASK = RAD50('MAPLOC')
  00192          .  .  .  .  .  ...FIN
  00193          .  .  .  .  .  ELSE
  00194          .  .  .  .  .  .  CALL STRTSB (MESS2)
  00195          .  .  .  .  .  .  CALL WAIT (2,2,MMM)
  00196          .  .  .  .  .  .  CALL STOPSB (MESS2)
  00197          .  .  .  .  .  .  NOCORD = 1
  00198          .  .  .  .  .  ...FIN
  00199          .  .  .  .  ...FIN
  00200          .  .  .  ...FIN
  00201          .  .  .  IF(NOCORD.EQ.0)
  00202          .  .  .  .  CALL DPW (IPNT0,0.,INODE,280.)
  00203          .  .  .  .  CALL REQUES (TASK,,IDS)
  00204          .  .  .  .  III = 0
  00205          .  .  .  .  IF(III.EQ.0) CALL EXIT
  00206          .  .  .  ...FIN
  00207          .  .  ...FIN
  00208          .  ...FIN
  00209          ...FIN
  00210          GO TO 10

         ---------------------------------------

  00211          TO RETRIEVE-CONTAMINATION-DATA
  00212          .  CALL DPFILE (4,'WELSPEC',2,,IPNT0)
  00213          .  CALL DPR (IPNT0,0.,INODE,280.)
  00214    C     .
  00215          .  QADR = INODE(61)
  00216          .  CALL DPR(ICHOR,QADR,IH,256.)
  00217    C     .
  00218          .  IOFF = (ICTYP-1)*3
```

```
(FLECS VERSION 22.46)      14-SEP-79   08100110  PAGE  00005
  00219          .  NPTS = IH(145+IOFF)
  00220          .  DKSAV= CTIADR(ICTYP)
  00221          .  DKSEX= CTXADR(ICTYP)
  00222          .  NPSEX= IH(146+IOFF)
  00223          .  INALOC=IH(147+IOFF)
  00224    C     .
  00225    C     .
  00226          .  WRDI = 2*NPTS
  00227          .  WI   = INALOC*256
  00228          .  DKT  = DKSAV+INALOC
  00229          .  DKM  = DKT+INALOC
  00230          .  CALL DPR(IPNT1,DKSAV,H,WRDI)
  00231          .  CALL DPR(IPNT1,DKT,T,WRDI)
  00232          .  IF (NPSEX .NE. 0)
  00233    C     .  .
  00234    C     READ EXTENDED AREA (IF ANY)
  00235    C     .  .
  00236          .  .  WRDE = 2*NPSEX
  00237          .  .  DKTE = DKSEX+2.
  00238          .  .  DKME = DKTE+2.
  00239          .  .  NPX  = NPTS+1
  00240          .  .  CALL DPR(IPNT1,DKSEX,H(NPX),WRDE)
  00241          .  .  CALL DPR(IPNT1,DKTE,T(NPX),WRDE)
  00242          .  ...FIN
  00243    C     REMOVE ANY MINUS VALUES
  00244          .  NTOT = NPTS+NPSEX
  00245          .  DO (I=1,NTOT)
  00246          .  .  H(I) = ABS(H(I))
  00247          .  ...FIN
  00248          ...FIN

        ----------------------------------------

  00249          TO CALCULATE-SCALE-FACTORS
  00250          .  IF (NVAR.EQ.0)
  00251          .  .  TMAX=0.0
  00252          .  .  TMIN=999999.
  00253          .  .  HMAX=0.0
  00254          .  .  HMIN=1.0E+20
  00255          .  .  DO (I=1,NTOT)
  00256          .  .  .  IF(T(I).GT.TMAX) TMAX=T(I)
  00257          .  .  .  IF(T(I).LT.TMIN) TMIN=T(I)
  00258          .  .  .  IF(H(I).GT.0.)
  00259          .  .  .  .  IF(H(I).GT.HMAX) HMAX=H(I)
  00260          .  .  .  .  IF(H(I).LT.HMIN) HMIN=H(I)
  00261          .  .  .  ...FIN
  00262          .  .  ...FIN
  00263          .  .  MINX=IFIX(TMIN/10000.)-1
  00264          .  .  MAXX=IFIX(TMAX/10000.)+1
  00265          .  .  AL1=ALOG10(HMIN)
  00266          .  .  IF(AL1.GE.0.) AL1=AL1+1.
  00267          .  .  AL2=ALOG10(HMAX)
  00268          .  .  IF(AL2.GE.0.) AL2=AL2+1.
  00269          .  .  MINY=IFIX(AL1)-1.
  00270          .  .  MAXY=IFIX(AL2)
  00271    C     .  .  WRITE(5,557) HMIN,HMAX,AL1,AL2,MINY,MAXY
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00006
  00272   557     .  .  FORMAT(4F10.4,2I5)
  00273           .  ...FIN
  00274           .  NVAR = 4
  00275           .  MINXOR = MINX
  00276           .  MAXXOR = MAXX
  00277           .  MINYOR = MINY
  00278           .  MAXYOR = MAXY
  00279           .  XDIF = (MAXX-MINX)+1
  00280           .  YDIF = (MAXY-MINY)
  00281           .  DSPX1 = 100.
  00282           .  DSPX2 = DSPX1 + 900.
  00283           .  DSPY1 = 100.
  00284           .  DSPY2 = DSPY1 + 800.
  00285           .  ...FIN
  00286   C

          ----------------------------------------

  00287           TO DRAW-THE-BACKGROUND-CHART
  00288           .  CALL SUBDF (BKGRND,100,IBG)
  00289           .  IBG = 2
  00290           .  CALL INTENS (5)
  00291           .  CALL APOS (IFIX(DSPX1),IFIX(DSPY1))
  00292           .  CALL BOX (IFIX(DSPX2-DSPX1),IFIX(DSPY2-DSPY1))
  00293           .  CALL CHRSCL (1)
  00294           .  CALL CHRTYP (0,1,0)
  00295           .  CALL APOS (30,346)
  00296           .  CALL TEXT ('CONCENTRATION  ')
  00297           .  GO TO(15,15,15,16,15,15,15,17,15,15,16,16,16,
  00298         1.       16,16,16,16,16,16,16,16,2050,16,16,16,15,
  00299         2.       16,16,18,19,20,15),ICTYP
  00300   15      .  CALL TEXT ('(PCI/ML)')
  00301           .  GO TO 2050
  00302   16      .  CALL TEXT ('(MG/L)')
  00303           .  GO TO 2050
  00304   17      .  CALL TEXT ('(NG/ML)')
  00305           .  GO TO 2050
  00306   18      .  CALL TEXT ('(PPM)')
  00307           .  GO TO 2050
  00308   19      .  CALL TEXT ('(UMHOS/CM3)')
  00309           .  GO TO 2050
  00310   20      .  CALL TEXT ('(UG/L)')
  00311           .  CALL CHRSCL (1)
  00312   2050    .  CALL APOS (460,60)
  00313           .  CALL CHRTYP (0,0,0)
  00314           .  CALL TEXT ('CALENDAR YEAR')
  00315   C       .
  00316           .  CALL CHRSCL (1)
  00317           .  CALL CHRTYP (0,0,0)
  00318           .  CALL APOS (250,950)
  00319           .  CALL TEXT ('CONCENTRATION HISTORY ---')
  00320           .  SELECT (ICTYP)
  00321           .  .  (1)  CALL TEXT ('TOTAL ALPHA')
  00322           .  .  (2)  CALL TEXT ('TOTAL BETA')
  00323           .  .  (3)  CALL TEXT ('TRITIUM')
  00324           .  .  (4)  CALL TEXT ('NITRATE')
```

```
(FLECS VERSION 22.46)     14-SEP-79   08100110  PAGE  00007
  00325          .  .  (5)  CALL TEXT ('STRONTIUM-90')
  00326          .  .  (6)  CALL TEXT ('CESIUM-137')
  00327          .  .  (7)  CALL TEXT ('COBALT-60')
  00328          .  .  (8)  CALL TEXT ('URANIUM-238')
  00329          .  .  (9)  CALL TEXT ('PLUTONIUM-239')
  00330          .  .  (10) CALL TEXT ('RUTHENIUM-106')
  00331          .  .  (11) CALL TEXT ('CHROMIUM')
  00332          .  .  (12) CALL TEXT ('FLUORIDE')
  00333          .  .  (13) CALL TEXT ('MAGNESIUM')
  00334          .  .  (14) CALL TEXT ('IRON')
  00335          .  .  (15) CALL TEXT ('PHOSPHATE')
  00336          .  .  (16) CALL TEXT ('CHLORIDE')
  00337          .  .  (17) CALL TEXT ('COPPER')
  00338          .  .  (18) CALL TEXT ('HARDNESS')
  00339          .  .  (19) CALL TEXT ('SOLIDS')
  00340          .  .  (20) CALL TEXT ('MANGANESE')
  00341          .  .  (21) CALL TEXT ('TOTAL ORGANIC CARBON')
  00342          .  .  (22) CALL TEXT ('PH')
  00343          .  .  (23) CALL TEXT ('SULFATE')
  00344          .  .  (24) CALL TEXT ('SODIUM')
  00345          .  .  (25) CALL TEXT ('CALCIUM')
  00346          .  .  (26) CALL TEXT ('TOTAL GAMMA')
  00347          .  .  (27) CALL TEXT ('BICARBONATE ION')
  00348          .  .  (28) CALL TEXT ('CARBONATE ION')
  00349          .  .  (29) CALL TEXT ('TOTAL POTASSIUM')
  00350          .  .  (30) CALL TEXT ('SPECIFIC CONDUCTIVITY')
  00351          .  .  (31) CALL TEXT ('BORON')
  00352          .  .  (32) CALL TEXT ('LOW ALPHA')
  00353          .  ...FIN
  00354          .  CALL CHRSCL (0)
  00355          .  CALL APOS (400,925)
  00356          .  CALL TEXT ('WELL DESIGNATION --')
  00357          .  CALL TEXT (WLDES,0,12)
  00358          .  CALL ENDSB(ISZ(1))
  00359          ...FIN

         ----------------------------------------

  00360          TO DISPLAY-X-AND-Y-LINES-FOR-CHART
  00361    C     .
  00362          .  CALL SUBDF (BKGLIN,900,IBL)
  00363          .  IBL = 2
  00364          .  CALL CHRSCL (0)
  00365          .  CALL INTENS (4)
  00366    C----DRAW Y LINES FOR CHART
  00367          .  DSYINC = (DSPY2-DSPY1)/YDIF
  00368          .  DO (JD = MINY,MAXY)
  00369          .  .  IYC = DSPY1 + FLOAT(JD-MINY)*DSYINC
  00370          .  .  CALL APOS (IFIX(DSPX1-30),IYC)
  00371          .  .  CALL TEXT ('10')
  00372          .  .  CALL RPOS (0,5)
  00373          .  .  CALL INMBR (JD,'(I2)')
  00374          .  .  DO ( YY = 1.,9.,1.)
  00375          .  .  .  IY = IYC + IFIX(ALOG10(YY)*DSYINC)
  00376          .  .  .  IF (JD.EQ.MAXY. AND .YY.GT.1.) GO TO 30
  00377          .  .  .  CALL APOS (IFIX(DSPX1),IY)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00008
  00378          .  .  .  CALL VEC  (IFIX(DSPX2),IY)
  00379          .  .  ...FIN
  00380    30    .  .  CONTINUE
  00381          .  ...FIN
  00382    C----DRAW X LINES FOR CHART
  00383          .  JDD = 1
  00384          .  DSXINC = (DSPX2-DSPX1)/XDIF
  00385          .  XADJ = DSXINC/2. + 7.
  00386          .  DO (JD = MINX,MAXX+1)
  00387          .  .  X = DSPX1 + FLOAT(JD-MINX+1)*DSXINC
  00388          .  .  IF (JD.LE.MAXX)
  00389          .  .  .  CALL APOS (IFIX(X-XADJ),IFIX(DSPY1-15))
  00390          .  .  .  CALL INMBR (JD,'(I2)')
  00391          .  .  ...FIN
  00392          .  .  IF(JD.LE.MAXX)
  00393          .  .  .  CALL APOS (IFIX(X),IFIX(DSPY1))
  00394          .  .  .  SELECT (JDD)
  00395          .  .  .  .  (1)
  00396          .  .  .  .  .  CALL RVEC (0,10)
  00397          .  .  .  .  .  JDD = 2
  00398          .  .  .  .  ...FIN
  00399          .  .  .  .  (2)
  00400          .  .  .  .  .  CALL RVEC (0,IFIX(DSPY2-DSPY1))
  00401          .  .  .  .  .  JDD = 1
  00402          .  .  .  .  ...FIN
  00403          .  .  .  ...FIN
  00404          .  .  ...FIN
  00405          .  ...FIN
  00406          .  CALL ENDSB(ISZ(2))
  00407          ...FIN

         ----------------------------------------

  00408          TO PLOT-DATA-POINTS-ON-GRAPH
  00409          .  XSCALE = (DSPX2-DSPX1)/(XDIF*365.)
  00410          .  IGR = 2
  00411          .  I1=0
  00412          .  I2=0
  00413          .  NDEL=0
  00414    C     .  WRITE(5,538) NTOT
  00415    538   .  FORMAT(' NTOT=',I5)
  00416    C     .  WRITE(5,539) (H(L),T(L),L=1,NTOT)
  00417    539   .  FORMAT(6F11.4)
  00418          .  DO (JK=1,NTOT)
  00419          .  .  Y = H(JK)
  00420          .  .  IYR  = T(JK)/10000.
  00421          .  .  MO   = T(JK)/100.-FLOAT(IYR)*100.
  00422          .  .  IDA  = T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
  00423    38    .  .  DAYS = (FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
  00424          .  .  X    = DAYS*XSCALE+DSPX1
  00425          .  .  WHEN (Y.LE.0.)
  00426          .  .  .  Y = DSPY1
  00427          .  .  .  NDEL=NDEL+1
  00428          .  .  .  IDEL(NDEL)=JK
  00429          .  .  ...FIN
  00430          .  .  ELSE
```

```
(FLECS VERSION 22.46)      14-SEP-79   08:00:10  PAGE  00009
  00431          .  .  .  Y = DSPY1 + (ALOG10(Y)-FLOAT(MINY))*DSYINC
  00432          .  .  ...FIN
  00433    C  CHECK FOR POINT WITHIN REGION
  00434          .  .  IF(X.GE.DSPX1. AND .X.LE.DSPX2)
  00435          .  .  .  IF(Y.GE.DSPY1. AND .Y.LE.DSPY2)
  00436          .  .  .  .  IF (NDEL.NE.0)
  00437    C  SEE IF POINT HAS BEEN DELETED
  00438          .  .  .  .  .  DO ( J=1,NDEL )
  00439          .  .  .  .  .  .  IF(IDEL(J).EQ.JK) GO TO 3820
  00440          .  .  .  .  .  ...FIN
  00441          .  .  .  .  ...FIN
  00442    C  NO
  00443          .  .  .  .  I1=I1+1
  00444          .  .  .  .  IT(I1) = X
  00445          .  .  .  .  IH(I1) = Y
  00446          .  .  .  .  GO TO 40
  00447    C  YES---STORE IN SECONDARY FILE
  00448    3820  .  .  .  .  I2=I2+1
  00449          .  .  .  .  IXDEL(I2)=X
  00450          .  .  .  .  IYDEL(I2)=Y
  00451          .  .  .  ...FIN
  00452          .  .  ...FIN
  00453    40    .  .  CONTINUE
  00454          .  ...FIN
  00455    C     .
  00456    C  PLOT DATA POINTS
  00457          .  CALL SUBDF (IGRAPH,4000,I40)
  00458          .  I40 = 2
  00459          .  CALL CHRSCL (0)
  00460          .  CALL INTENS (4)
  00461          .  DO (I=1,I1)
  00462          .  .  CALL APOS (IT(I)-3,IH(I)-4)
  00463          .  .  CALL TEXT ('*')
  00464          .  ...FIN
  00465          .  IF (JUIN)
  00466          .  .  DO (I=1,I1)
  00467          .  .  .  WHEN (I.EQ.1) CALL APOS (IT(I),IH(I))
  00468          .  .  .  ELSE
  00469          .  .  .  .  CALL VEC (IT(I),IH(I))
  00470          .  .  .  ...FIN
  00471          .  .  ...FIN
  00472          .  ...FIN
  00473          .  IF(I2.NE.0)
  00474    C  PLOT DELETED POINTS
  00475          .  .  DO (I=1,I2)
  00476          .  .  .  CALL APOS (IXDEL(I)-3,IYDEL(I)-4)
  00477          .  .  .  CALL TEXT ('0')
  00478          .  .  ...FIN
  00479          .  ...FIN
  00480          .  CALL ENDSB(ISZ(4))
  00481          .  CALL STRTSB (IGRAPH)
  00482          ...FIN

         ----------------------------------------

  00483          TO SETUP-LIGHT-BUTTONS
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00010
  00484    C      .
  00485           .   CALL SUBOF (LBUT1,350,0)
  00486           .   CALL AREA (1)
  00487           .   CALL CHRSCL (0)
  00488           .   CALL INTENS (4)
  00489           .   CALL APOS (0,700)
  00490           .   CALL TEXT ('  LIGHT PEN OPTIONS')
  00491           .   CALL INTENS (7)
  00492           .   CALL APOS (0,685)
  00493           .   CALL BOX (150,40)
  00494           .   CALL APOS (5,690)
  00495           .   CALL BOX (140,30)
  00496           .   CALL APOS (30,625)
  00497           .   CALL TEXT ('SCALE CHANGE',-3)
  00498           .   CALL RVEC (84,0)
  00499           .   CALL MENU (590,30,1,'CONCENTRATION SCALE','TIME SCALE',
  00500         1.    'ORIGINAL SCALE')
  00501           .   CALL AREA (1)
  00502           .   CALL APOS (30,450)
  00503           .   CALL TEXT ('OUTPUT DEVICES',-3)
  00504           .   CALL RVEC (90,0)
  00505           .   CALL MENU (415,30,4,'CALCOMP--HANFORD,MAP','CALCOMP--REGION MAP',
  00506         1.    'GOULD')
  00507           .   CALL AREA (1)
  00508           .   CALL APOS (30,300)
  00509           .   CALL TEXT ('SPECIAL FUNCTIONS',-3)
  00510           .   CALL RVEC (120,0)
  00511           .   CALL MENU (265,30,7,'JOIN POINTS','UNJOIN POINTS',
  00512         1.    'LEAST SQUARES FIT')
  00513           .   CALL AREA (1)
  00514           .   CALL APOS (30,150)
  00515           .   CALL TEXT ('PROGRAM CONTROLS',-3)
  00516           .   CALL RVEC (110,0)
  00517           .   CALL MENU (115,30,20,'RETURN TO CIRMIS CONTROL',
  00518         1.    'RETURN TO WELL SELECT','DISPLAY MAP LOCATION')
  00519           .   CALL AREA (0)
  00520           .   CALL ENDSB(ISZ(5))
  00521    C      .
  00522    C *** OUTPUT MESSAGES
  00523           .   CALL SUBOF (MESSY1,30,1)
  00524           .   CALL CHRSCL (0)
  00525           .   CALL INTENS (7)
  00526           .   CALL BLINK (1)
  00527           .   CALL APOS (0,30)
  00528           .   CALL TEXT ('ENTER MINIMUM Y EXPONENT (I4)')
  00529           .   CALL BLINK (0)
  00530           .   CALL ENDSB(ISZ(6))
  00531    C      .
  00532           .   CALL SUBOF (MESSY2,30,1)
  00533           .   CALL CHRSCL (0)
  00534           .   CALL INTENS (7)
  00535           .   CALL BLINK (1)
  00536           .   CALL APOS (0,30)
  00537           .   CALL TEXT ('ENTER MAXIMUM Y EXPONENT (I4)')
  00538           .   CALL BLINK (0)
  00539           .   CALL ENDSB (ISZ(7))
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:00:10  PAGE  00011
  00540   C       .
  00541           .  CALL SUBDF (MESSX1,30,1)
  00542           .  CALL CHRSCL (0)
  00543           .  CALL INTENS (7)
  00544           .  CALL BLINK (1)
  00545           .  CALL APOS (0,30)
  00546           .  CALL TEXT ('ENTER MINIMUM TIME IN YEARS (I4)')
  00547           .  CALL BLINK (0)
  00548           .  CALL ENDSB(ISZ(8))
  00549   C       .
  00550           .  CALL SUBDF (MESSX2,30,1)
  00551           .  CALL CHRSCL (0)
  00552           .  CALL INTENS (7)
  00553           .  CALL BLINK (1)
  00554           .  CALL APOS (0,50)
  00555           .  CALL TEXT ('ENTER MAXIMUM TIME IN YEARS (I4)')
  00556           .  CALL BLINK (0)
  00557           .  CALL ENDSB(ISZ(9))
  00558   C       .
  00559           .  CALL SUBDF (MESS2,40,1)
  00560           .  CALL CHRSCL (0)
  00561           .  CALL AREA (1)
  00562           .  CALL BLINK (1)
  00563           .  CALL APOS (0,200)
  00564           .  CALL TEXT ('COORDINATES NOT AVAILABLE FOR THIS WELL')
  00565           .  CALL BLINK (0)
  00566           .  CALL AREA (0)
  00567           .  CALL ENDSB(ISZ(10))
  00568           ...FIN
  00569           END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00249 CALCULATE-SCALE-FACTORS
                00072

          00360 DISPLAY-X-AND-Y-LINES-FOR-CHART
                00075  00106  00119  00129

          00287 DRAW-THE-BACKGROUND-CHART
                00073

          00408 PLOT-DATA-POINTS-ON-GRAPH
                00077  00107  00120  00130  00170  00174

          00211 RETRIEVE-CONTAMINATION-DATA
                00071

          00483 SETUP-LIGHT-BUTTONS
                00062


          (FLECS VERSION 22.46)
```

```
00570          SUBROUTINE GRECHO(KEYBUF,NCHAR,IX,IY,ICS,IKB)
00571    C
00572          BYTE KEYBUF(80)
00573          INTEGER DEL,CR
00574          COMMON/DSP/ MAIN(100),KEYOUT(100)
00575    C
00576          DEL = "177
00577          CR = "15
00578          CALL SUBDF(KEYOUT,100,IKB)
00579          IKB = 2
00580          CALL INTENS (4)
00581          CALL CHRSCL (ICS)
00582          CALL APOS (IX,IY)
00583          CALL TEXT ('>SE')
00584          CALL ENDSB
00585    C
00586          NCHAR = 0
00587    C
00588    10    CALL GRATTN (1,ITYP,'KB')
00589          IF (ITYP.NE.'KB') GO TO 10
00590          CALL KEYBRD (KEY)
00591          IF (KEY.EQ.CR) GO TO 30
00592          IF (NCHAR.EQ.80) GO TO 30
00593          IF (NCHAR.LT.0) GO TO 10
00594          NCHAR = NCHAR+1
00595          KEYBUF(NCHAR) = KEY
00596          IF (KEY.EQ.DEL)
00597          .  NCHAR = NCHAR-2
00598          ...FIN
00599          CALL SUBDF(KEYOUT,100,IKB)
00600          CALL INTENS (4)
00601          CALL CHRSCL (ICS)
00602          CALL APOS(IX,IY)
00603          CALL TEXT (KEYBUF,0,NCHAR)
00604          CALL TEXT ('>SE')
00605          CALL ENDSB
00606          GO TO 10
00607    30    RETURN
00608          END
```

(FLECS VERSION 22.46)

```
00609          SUBROUTINE TRAP
00610          STOP 'UNEXPECTED INTERRUPT RECIEVED'
00611          END
```

(FLECS VERSION 22.46)

WLGSCP.FLX

DRILLERS LOG (SCOPE)

```
(FLECS VERSION 22,46)     14-SEP-79   08:01:00  PAGE  00001

          ----------------------------------------

  00001   C ***************  [351,101]WLGSCP.FLX  ***************
  00002   C
  00003   C
  00004   C  PROGRAM ID:  SRV-55-WEL-3-0 ABBREVIATED WELL NAME VERSION
  00005   C
  00006   C  PROGRAM DESCRIPTION:
  00007   C     THIS PROGRAM OUTPUTS DRILLERS LOGS TO VS60 DISPLAY.
  00008   C     DATA OUTPUT: STRATIFICATION, SOIL TYPES, AND COMMENTS,
  00009   C     "HANFORD RESERVATION"
  00010   C
  00011   C  DATA FILES:
  00012   C        NAME                        LUN          TYPE          ACCESS
  00013   C        FILE0(DM0)---FRMTWLG         3           RAN             R
  00014   C        FILE0(UK1)---WELSPEC         4           RAN            R/W
  00015   C
  00016   C
  00017   C
  00018   C
  00019   C  BATTELLE MEMORIAL INSTITUTE
  00020   C  PACIFIC NORTHWEST LABORATORIES
  00021   C  WATER & LAND RESOURCES DEPT.
  00022   C
  00023   C  AUTHOR(S):    DR FRIEDRICHS
  00024   C
  00025   C
  00026   C  DATE:         INITIAL VERSION   FEBRUARY 1978
  00027   C                CURRENT VERSION   MARCH 1978
  00028   C
  00029   C
  00030   C
  00031         BYTE WLDES
  00032         BYTE STRNG1(720),BLANK
  00033         LOGICAL FIRST,READY,DONE
  00034   C
  00035         DIMENSION SOILS(180),STRING(180),SLAST(180)
  00036         DIMENSION EXPL(100),ELAST(100),INODE(280)
  00037         DIMENSION NDUM(3000),NBUF(256)
  00038         DIMENSION DDATE(2),T(2),TC(2)
  00039   C
  00040         COMMON/HDR/ WLDES(12),IDUM1(10),CASE,IDUM2(21),IBLK,NSTRT,NWRD,
  00041        1          IDUM3(187),OWN(6),DRL(6),IDUM4(17)
  00042         COMMON/BLK2/ BUF(1500)
  00043         COMMON/DSP/ MAIN(100),IBOX(15), LCASE(200), LOGDSP(5000),
  00044        1           LBUT(75),MES81(55),NCONT(30)
  00045   C
  00046         EQUIVALENCE (BUF(1),NDUM(1)),(NRETRN,INODE(258))
  00047         EQUIVALENCE (STRING(1),STRNG1(1))
  00048         EQUIVALENCE (INODE(1),WLDES(1))
  00049   C
  00050         DATA BLANKS /'    '/
  00051         DATA BLANK  /' '/
  00052   C
  00053   C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00002
  00054    C
  00055           CALL MAINOF (MAIN,100,0,1)
  00056           CALL CLKOFF
  00057    C
  00058           CALL OPFILE (4,'WELSPEC',2,,IPNT0)
  00059           CALL OPR (IPNT0,0,,INODE,280,)
  00060           CALL OPFILE (3,'FRMTWLG',8000,,IPNT1)
  00061    C
  00062    C
  00063           RETRIEVE-DATA
  00064    C      WRITE(5,577) NTOT
  00065    577    FORMAT(' NTOT=',I5)
  00066           READY = .FALSE.
  00067           DONE = .FALSE.
  00068    C
  00069    C
  00070           CALL SDINIT (2)
  00071           CALL SDJOYF
  00072           START-INITIAL-DISPLAY
  00073    C
  00074    C
  00075    10     CALL GRATTN (0,ITYP,'LP')
  00076           CALL SDJOYR (IX,JOYY)
  00077           IF (READY)
  00078           .  IF (JOYY.NE.IYLAST)
  00079           .  .  IYLAST = JOYY
  00080           .  .  IY81 = 700. - FLOAT(IYSTRT-JOYY)*FACJOY*DSCALE
  00081           .  .  IY82 = 3072. - FLOAT(IYSTRT-JOYY)*FACJOY*DSCALE
  00082           .  .  CALL CHGGE (1,0,IY81)
  00083           .  .  CALL CHGGE (2,0,-IY82)
  00084           .  ...FIN
  00085           ...FIN
  00086           IF (ITYP.EQ.'LP')
  00087           .  CALL LIPEN (ID,ITIP)
  00088           .  IF (ITIP.EQ.1)
  00089           .  .  SELECT (ID)
  00090           .  .  .  (1)
  00091           .  .  .  .  NRETRN = 0
  00092           .  .  .  .  TASK = RAD50 ('CIRMIS')
  00093           .  .  .  ...FIN
  00094           .  .  .  (2)
  00095           .  .  .  .  NRETRN = 1
  00096           .  .  .  .  TASK = RAD50 ('WELSEL')
  00097           .  .  .  ...FIN
  00098           .  .  .  (10)
  00099           .  .  .  .  IF (.NOT.DONE)
  00100    C      .  .  .  .  .  WRITE(5,511) ID
  00101    511    .  .  .  .  .  FORMAT(' ID = ',I5)
  00102           .  .  .  .  .  CALL STOPSR (MESS1,NCONT)
  00103           .  .  .  .  .  CALL WAIT (1,2,MMM)
  00104           .  .  .  .  .  SETUP-DISPLAY-FILES
  00105           .  .  .  .  .  IYSTRT = JOYY
  00106           .  .  .  .  .  IYLAST = JOYY
  00107           .  .  .  .  .  FACJOY = 800./FLOAT(JOYY)
  00108           .  .  .  .  .  READY = .TRUE.
  00109           .  .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00003
  00110          .  .  .  ...FIN
  00111          .  .  ...FIN
  00112          .  .  IF (ID.EQ.1. OR .ID.EQ.2)
  00113          .  .  .  CALL DPW (IPNT0,0.,INODE,280.)
  00114          .  .  .  CALL REQUES (TASK,,IDS)
  00115          .  .  .  III = 0
  00116          .  .  .  IF (III.EQ.0) CALL EXIT
  00117          .  .  ...FIN
  00118          .  ...FIN
  00119          ...FIN
  00120          GO TO 10
  00121    C

         ------------------------------------

  00122          TO RETRIEVE-DATA
  00123          .  NWORD=NWRD
  00124          .  KBLK=IBLK
  00125          .  JB=NSTRT
  00126          .  STRT=KBLK
  00127          .  CALL DPR(IPNT1,STRT,NBUF,256.)
  00128          .  DO (J=1,NWORD)
  00129          .  .  JB=JB+1
  00130          .  .  NDUM(J)=NBUF(JB)
  00131          .  .  IF (JB.GE.256)
  00132          .  .  .  KBLK=KBLK+1
  00133          .  .  .  STRT=KBLK
  00134          .  .  .  CALL DPR(IPNT1,STRT,NBUF,256.)
  00135          .  .  .  JB=0
  00136          .  .  ...FIN
  00137          .  ...FIN
  00138    C     .
  00139          .  NTOT=BUF(4)
  00140          .  DDATE(1)=BUF(5)
  00141          .  DDATE(2)=BUF(6)
  00142          .  TOTDEP=BUF(7)
  00143          .  N=7
  00144          ...FIN

         ------------------------------------

  00145          TO START-INITIAL-DISPLAY
  00146          .  CALL SUBDF (MESS1,55,0)
  00147          .  CALL CHRSCL (1)
  00148          .  CALL INTENS (2)
  00149          .  CALL APOS (230,600)
  00150          .  CALL TEXT ('PULL THE JOYSTICK ALL THE WAY FORWARD THEN',-30)
  00151          .  CALL TEXT ('SELECT "CONTINUE" WITH THE LIGHT PEN.')
  00152          .  CALL ENDSB
  00153    C     .
  00154          .  CALL SUBDF (NCONT,30)
  00155          .  CALL INTENS (2)
  00156          .  CALL CHRSCL (1)
  00157          .  CALL NAME (10)
  00158          .  CALL APOS (450,500)
  00159          .  CALL TEXT ('CONTINUE')
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00004
  00160          .  CALL APOS (440,490)
  00161          .  CALL BOX (132,35)
  00162          .  CALL INTENS (7)
  00163          .  CALL APOS (435,485)
  00164          .  CALL BOX (142,45)
  00165          .  CALL ENDSR
  00166          ...FIN

         ----------------------------------------

  00167          TO SETUP-DISPLAY-FILES
  00168          .  CALL SUBDF (LCASE,200,0)
  00169          .  DSCAL1 = 100./145.
  00170          .  CALL AREA (1)
  00171          .  CALL INTENS (3)
  00172          .  CALL CHRSCL (0)
  00173          .  DO (DDD=0,,TOTDEP,50.)
  00174          .  .  IY = 783.-DDD*DSCAL1
  00175          .  .  CALL APOS (23,IY)
  00176          .  .  CALL RVEC (10,0)
  00177          .  .  IDP = DDD
  00178          .  .  CALL INMAR (IDP,'(I4)')
  00179          .  ...FIN
  00180          .  CALL INTENS (7)
  00181          .  CALL APOS (0,783)
  00182          .  CALL RVEC (23,0)
  00183          .  CALL RPOS (7,0)
  00184          .  CALL RVEC (60,0)
  00185          .  IY = TOTDEP*DSCAL1 + 3
  00186          .  CALL APOS (23,786)
  00187          .  CALL RVEC (0,-IY)
  00188          .  CALL RPOS (7,0)
  00189          .  CALL RVEC (0,IY)
  00190          .  CALL AREA (0)
  00191          .  CALL ENDSR
  00192    C     .
  00193          .  CALL SUBDF (IBOX,15,0)
  00194          .  CALL AREA (1)
  00195          .  CALL INTENS (6)
  00196          .  CALL POINTR (1)
  00197          .  CALL APOS (0,700)
  00198          .  CALL RVEC (100,0)
  00199          .  CALL RVEC (0,100)
  00200          .  CALL RVEC (-100,0)
  00201          .  CALL RVEC (0,-100)
  00202          .  CALL AREA (0)
  00203          .  CALL ENDSR
  00204    C     .
  00205          .  CALL SUBDF (LBUT,75)
  00206          .  CALL AREA (1)
  00207          .  CALL CHRSCL (0)
  00208          .  CALL INTENS (4)
  00209          .  CALL APOS (15,950)
  00210          .  CALL TEXT ('LIGHT PEN OPTIONS')
  00211          .  CALL APOS (5,940)
  00212          .  CALL BOX (140,30)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00005
  00213           .  CALL INTENS (7)
  00214           .  CALL APOS (0,935)
  00215           .  CALL BOX (150,40)
  00216           .  CALL MENU (900,30,1,'RETURN TO MONITOR',
  00217         1 .  'RETURN TO WELL SELECT')
  00218           .  CALL AREA (0)
  00219           .  CALL ENDSB
  00220    C      .
  00221           .  CALL SUBDF(LOGDSP,5000,0)
  00222           .  IYBOT = 3072
  00223           .  IYTOP = IYBOT+850
  00224           .  CALL POINTR (2)
  00225           .  CALL WINDOW (0,IYBOT)
  00226           .  CALL CHRSCL (3)
  00227           .  CALL INTENS (7)
  00228           .  CALL APOS (340,IYBOT+1000)
  00229           .  CALL TEXT ('DRILLERS LOG')
  00230           .  CALL CHRSCL (1)
  00231           .  CALL INTENS (5)
  00232           .  CALL APOS (300,4050)
  00233           .  CALL TEXT ('WELL DESIGNATION = ')
  00234           .  CALL TEXT (WLDES,0,12)
  00235           .  IY = IYTOP+5
  00236           .  CALL CHRSCL (1)
  00237           .  CALL APOS (14,IY)
  00238           .  CALL TEXT ('STRATIGRAPHY')
  00239           .  CALL APOS (400,IY)
  00240           .  CALL TEXT ('MATERIAL PENETRATED')
  00241           .  CALL CHRSCL (0)
  00242           .  CALL INTENS (4)
  00243           .  CALL APOS (0,IYTOP+100)
  00244           .  CALL TEXT ('DATE:',-15)
  00245           .  CALL TEXT ('CASING:',-15)
  00246           .  CALL TEXT ('DRILLER:',-15)
  00247           .  CALL TEXT ('FORMAN:')
  00248    C      .
  00249    C  DRAW CASING.  SET SCALE AT 145FT = 1023 DISPLAY UNITS
  00250           .  DSCALE = 1023./145.
  00251           .  IBOT = TOTDEP*DSCALE
  00252           .  CALL INTENS (7)
  00253           .  CALL APOS (0,IYTOP)
  00254           .  CALL RVEC (240,0)
  00255           .  CALL RPOS (40,0)
  00256           .  CALL RVEC (520,0)
  00257    C      .
  00258           .  IY1 = IYTOP-IBOT
  00259           .  IY2 = IYTOP+25
  00260           .  CALL APOS (240,IY1)
  00261           .  CALL SPLTVC (IY1,IY2)
  00262           .  CALL RVEC (4,0)
  00263           .  CALL SPLTVC (IY2,IY1)
  00264           .  CALL RPOS (32,0)
  00265           .  CALL SPLTVC (IY1,IY2)
  00266           .  CALL RVEC (4,0)
  00267           .  CALL SPLTVC (IY2,IY1)
  00268           .  CALL INTENS (4)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00006
  00269   C      .
  00270   C      .
  00271          .  PREVD=0.0
  00272          .  I1 = 0
  00273          .  NS=0
  00274          .  NEX=0
  00275          .  ITYPE=0
  00276          .  FIRST = .TRUE.
  00277          .  DONE = .FALSE.
  00278   C      .  WRITE(5,512)
  00279   512    .  FORMAT(' SELECT')
  00280   C      .
  00281   C      .
  00282   C  ****** READ DATA FROM BUFFER ******
  00283   C      .
  00284          .  REPEAT UNTIL (DONE)
  00285          .  .  N=N+1
  00286   C      .  .  WRITE(5,539) N,ITYPE,DONE,BUF(N)
  00287   539    .  .  FORMAT(3I5,F10.2)
  00288          .  .  I1=BUF(N)
  00289          .  .  SELECT (I1)
  00290          .  .  .  (1)
  00291   C  CARD TYPE 1---HEADER INFO
  00292          .  .  .  .  IF (ITYPE.NE.0)
  00293          .  .  .  .  .  SELECT (ITYPE)
  00294          .  .  .  .  .  .  (1) PLOT-HEADER
  00295          .  .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
  00296          .  .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
  00297          .  .  .  .  .  ...FIN
  00298          .  .  .  .  ...FIN
  00299          .  .  .  .  ITYPE=1
  00300          .  .  .  ...FIN
  00301   C      .  .  .
  00302   C  CARD TYPE 2---STRATIFICATION
  00303          .  .  .  (2)
  00304          .  .  .  .  IF (ITYPE.NE.0)
  00305          .  .  .  .  .  SELECT (ITYPE)
  00306          .  .  .  .  .  .  (1) PLOT-HEADER
  00307          .  .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
  00308          .  .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
  00309          .  .  .  .  .  ...FIN
  00310          .  .  .  .  ...FIN
  00311          .  .  .  .  ITYPE=2
  00312          .  .  .  .  N=N+1
  00313          .  .  .  .  TIM2=BUF(N)
  00314          .  .  .  .  N=N+1
  00315          .  .  .  .  DEPTH=BUF(N)
  00316          .  .  .  .  N=N+1
  00317          .  .  .  .  ISAMP=BUF(N)
  00318          .  .  .  .  N=N+1
  00319          .  .  .  .  ICONT=BUF(N)
  00320          .  .  .  .  N=N+1
  00321          .  .  .  .  NEXPL=BUF(N)
  00322          .  .  .  .  IF(NEXPL.LT.0) WRITE(5,581) NEXPL
  00323   581    .  .  .  .  FORMAT(' NEXPL =',I6)
  00324          .  .  .  .  IF (NEXPL.NE.0)
```

```
(FLECS VERSION 22.46)      14-SEP-79   08101100  PAGE  00007
  00325          .  .  .  .  .  DO (J=1,NEXPL)
  00326          .  .  .  .  .  .  N=N+1
  00327          .  .  .  .  .  .  EXPL(J)=BUF(N)
  00328          .  .  .  .  .  ...FIN
  00329          .  .  .  .  ...FIN
  00330          .  .  .  ...FIN
  00331    C     .  .  .
  00332    C  CARD TYPE 3---SOILS
  00333          .  .  .  (3)
  00334          .  .  .  .  IF (ITYPE.NE.0)
  00335          .  .  .  .  .  SELECT (ITYPE)
  00336          .  .  .  .  .  .  (1) PLOT-HEADER
  00337          .  .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
  00338          .  .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
  00339          .  .  .  .  .  ...FIN
  00340          .  .  .  .  ...FIN
  00341          .  .  .  .  ITYPE=3
  00342          .  .  .  .  N=N+1
  00343          .  .  .  .  TIM3=BUF(N)
  00344          .  .  .  .  N=N+1
  00345          .  .  .  .  DEPTH3=BUF(N)
  00346          .  .  .  .  N=N+1
  00347          .  .  .  .  NSOIL=BUF(N)
  00348          .  .  .  .  IF(NSOIL.LT.0) WRITE(5,582) NSOIL
  00349    582   .  .  .  .  FORMAT(' NSOIL =',I6)
  00350          .  .  .  .  IF (NSOIL.NE.0)
  00351          .  .  .  .  .  DO (J=1,NSOIL)
  00352          .  .  .  .  .  .  N=N+1
  00353          .  .  .  .  .  .  SOILS(J)=BUF(N)
  00354          .  .  .  .  .  ...FIN
  00355          .  .  .  .  ...FIN
  00356          .  .  .  ...FIN
  00357    C     .  .  .
  00358    C  CARD TYPE 4---COMMENTS
  00359          .  .  .  (4)
  00360          .  .  .  .  IF (ITYPE.NE.0)
  00361          .  .  .  .  .  SELECT (ITYPE)
  00362          .  .  .  .  .  .  (1) PLOT-HEADER
  00363          .  .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
  00364          .  .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
  00365          .  .  .  .  .  ...FIN
  00366          .  .  .  .  ...FIN
  00367          .  .  .  .  ITYPE=4
  00368          .  .  .  .  N = N+4
  00369          .  .  .  .  NCOM=BUF(N)
  00370          .  .  .  .  N = N+NCOM
  00371          .  .  .  ...FIN
  00372    C     .  .  .
  00373    C  CARD TYPE 5---END OF DATA (PLOT REMAINING DATA IN BUFFER)
  00374          .  .  .  (5)
  00375          .  .  .  .  IF (ITYPE.NE.0)
  00376          .  .  .  .  .  SELECT (ITYPE)
  00377          .  .  .  .  .  .  (1) PLOT-HEADER
  00378          .  .  .  .  .  .  (2) PLOT-LAST-STRATIGRAPHY
  00379          .  .  .  .  .  .  (3) PLOT-LAST-SOIL-TYPE
  00380          .  .  .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00009
  00431   72      .  IY  = IYTOP - (DEPTHE+DEPTH)*DSCALE/2.
  00432           .  IYT = IYTOP - (DEPTHE)*DSCALE - 3
  00433           .  IYB = IYTOP - (DEPTH)*DSCALE + 3
  00434           .  CALL APOS (235,IYT)
  00435           .  CALL RVEC (-35,0)
  00436           .  CALL SPLTVC (IYT,IYB)
  00437           .  CALL RVEC (35,0)
  00438           .  CALL APOS (150,IY)
  00439           .  CALL RVEC (50,0)
  00440           .  LE=0
  00441           .  DO (J=1,NEX)
  00442           .  .  IF (ELAST(J),NE,BLANKS)
  00443           .  .  .  LE=LE+1
  00444           .  .  .  STRING(J)=ELAST(J)
  00445           .  .  ...FIN
  00446           .  ...FIN
  00447           .  LEN=LE*4
  00448           .  CALL CHRSCL (0)
  00449           .  CALL APOS (0,IY)
  00450           .  CALL TEXT (STRING,0,LEN)
  00451   74      .  IF (NEXPL,NE,0)
  00452           .  .  DO (J=1,NEXPL)
  00453           .  .  .  ELAST(J)=EXPL(J)
  00454           .  .  ...FIN
  00455           .  ...FIN
  00456           .  NEX=NEXPL
  00457   77      .  DEPTHE=DEPTH
  00458           ...FIN

          ----------------------------------------

  00459           TO PLOT-LAST-SOIL-TYPE
  00460   C       .
  00461   C    CARD TYPE 3
  00462   C       .
  00463   C  CHECK FOR SOIL
  00464           .  IF (NS,EQ,0) GO TO 84
  00465           .  IF(NS,NE,NSOIL) GO TO 82
  00466           .  DO (J=1,NSOIL)
  00467           .  .  IF(SLAST(J),NE,SOILS(J)) GO TO 82
  00468           .  ...FIN
  00469   C    SAME SOIL
  00470           .  TIMEL=TIM2
  00471           .  GO TO 87
  00472   C  NEW SOIL--PLOT LAST ONE
  00473   82      .  IY = IYTOP -DEPTHL*DSCALE
  00474           .  CALL CHRSCL (0)
  00475           .  CALL APOS (240,IY)
  00476           .  CALL RVEC (45,0)
  00477           .  CALL FNMBR (DEPTHL,'(F6,1)')
  00478           .  IY = IYTOP - (PREVD+DEPTHL)*DSCALE/2.
  00479           .  LS=0
  00480           .  DO (J=1,NS)
  00481           .  .  IF (SLAST(J),NE,BLANKS)
  00482           .  .  .  LS=LS+1
  00483           .  .  .  STRING(LS)=SLAST(J)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101100  PAGE  00008
  00381          .  .  .  .  ...FIN
  00382          .  .  .  .  ITYPE=5
  00383          .  .  .  .  DONE = .TRUE.
  00384    C     .  .  .  .  WRITE(5,541) ITYPE,DONE
  00385    541   .  .  .  .  FORMAT(' ITYPE,DONE=',2I5)
  00386          .  .  .  ...FIN
  00387          .  .  ...FIN
  00388    C     .  .  WRITE(5,542) ITYPE,DONE
  00389    542   .  .  FORMAT('   *** ITYPE,DONE=',2I5)
  00390          .  ...FIN
  00391          .  NEXPL = 0
  00392          .  DEPTH = TOTDEP
  00393          .  PLOT-LAST-STRATIGRAPHY
  00394          .  CALL ENDSB(IIIZ)
  00395    C     .  WRITE(5,566) IIIZ
  00396    566   .  FORMAT(' SIZE =',I5)
  00397          .  CALL STRTSB (LOGOSP)
  00398          ...FIN

         ----------------------------------------

  00399          TO PLOT-HEADER
  00400    C   CARD TYPE 1
  00401    C     .
  00402          .  IF (FIRST)
  00403          .  .  FIRST = .FALSE.
  00404          .  .  IY = IYTOP+100
  00405          .  .  CALL CHRSCL (0)
  00406          .  .  CALL APOS (65,IY)
  00407          .  .  CALL TEXT (DDATE,-15,6)
  00408          .  .  CALL FNMBR (CASE,'(F7.2)',-15)
  00409          .  .  DO (J=1,6)
  00410          .  .  .  STRING(J)=DRL(J)
  00411          .  .  ...FIN
  00412          .  .  CALL TEXT (STRING,-15,24)
  00413          .  .  DO (J=1,6)
  00414          .  .  .  STRING(J)=OWN(J)
  00415          .  .  ...FIN
  00416          .  .  CALL TEXT (STRING,-15,24)
  00417          .  ...FIN
  00418          ...FIN

         ----------------------------------------

  00419          TO PLOT-LAST-STRATIGRAPHY
  00420    C     .
  00421    C   CARD TYPE 2
  00422    C     .
  00423          .  IF(NEX.EQ.0) GO TO 74
  00424          .  IF(NEX.NE.NEXPL) GO TO 72
  00425          .  DO (J=1,NEX)
  00426          .  .  IF(ELAST(J).NE.EXPL(J)) GO TO 72
  00427          .  ...FIN
  00428    C  SAME
  00429          .  GO TO 77
  00430    C  NEW STRATA--PLOT LAST ONE
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:01:00  PAGE  00010
  00484          .  .  ...FIN
  00485          .  ...FIN
  00486          .  LEN=LS*4
  00487          .  CALL APOS (360,IY)
  00488          .  CALL TEXT (STRING,0,LEN)
  00489          .  PREVD=DEPTHL
  00490    84    .  CONTINUE
  00491          .  IF (NSOIL.NE.0)
  00492          .  .  DO (J=1,NSOIL)
  00493          .  .  .  SLAST(J)=SOILS(J)
  00494          .  .  ...FIN
  00495          .  ...FIN
  00496          .  NS=NSOIL
  00497    87    .  DEPTHL=DEPTH3
  00498          ...FIN
  00499          END

         ----------------------------------------

               PROCEDURE CROSS-REFERENCE TABLE

         00399 PLOT-HEADER
               00294  00306  00336  00362  00377

         00459 PLOT-LAST-SOIL-TYPE
               00296  00308  00338  00364  00379

         00419 PLOT-LAST-STRATIGRAPHY
               00295  00307  00337  00363  00378  00393

         00122 RETRIEVE-DATA
               00063

         00167 SETUP-DISPLAY-FILES
               00104

         00145 START-INITIAL-DISPLAY
               00072


         (FLECS VERSION 22.46)

         ----------------------------------------

  00500          SUBROUTINE SPLTVC (IY1,IY2)
  00501    C
  00502          LOGICAL DONE
  00503    C
  00504          COMMON/DSP/ MAIN(100),IBOX(15), LCASE(200), LOGDSP(5000)
  00505    C
  00506          YLAST = IY1
  00507          YY = IY2
  00508          SIGNY = 1.
  00509          IF (YY.LT.YLAST) SIGNY = -1.
  00510          YFACT = 1023.*SIGNY
  00511          DONE = .FALSE.
```

```
(FLECS VERSION 22,46)     14-SEP-79   00101000  PAGE  00011
  00512          REPEAT UNTIL (DONE)
  00513          .  YDIF = ABS (YY-YLAST)
  00514          .  WHEN (YDIF.GT.1023.)
  00515          .  .  Y2 = YLAST+YFACT
  00516          .  .  IIY = Y2 - YLAST
  00517          .  .  YLAST = Y2
  00518          .  .  CALL RVEC (0,IIY)
  00519          .  ...FIN
  00520          .  ELSE
  00521          .  .  IIY = FLOAT(IY2) - YLAST
  00522          .  .  CALL RVEC (0,IIY)
  00523          .  .  DONE = .TRUE.
  00524          .  ...FIN
  00525          ...FIN
  00526          RETURN
  00527          END

          (FLECS VERSION 22,46)

          ------------------------------------
```

TMPSCP.FLX

TEMPERATURE (SCOPE)

```
(FLECS VERSION 22.46)     14-SEP-79   08101143  PAGE  00001

        ----------------------------------------

00001   C ******************** [351,101]TMPSCP.FLX ********************
00002   C
00003   C
00004   C  PROGRAM ID: SRV-55-WEL-10-0 ABBREVIATED WELL DESIG. VERSION
00005   C
00006   C  PROGRAM DESCRIPTION:
00007   C     THIS PROGRAM WAS DESIGNED TO RETRIEVE WELL TEMPERATURE DATA
00008   C     FROM THE CIRMIS DATA BANK AND DISPLAY A PLOT OF THE DATA.
00009   C
00010   C  DATA FILES:
00011   C        NAME                        LUN          TYPE          ACCESS
00012   C        FILEQ(DB0)---FRMTTMP         3           RAN            R
00013   C        FILEQ(DK1)---WELSPEC         4           RAN            R/W
00014   C
00015   C
00016   C
00017   C
00018   C  BATTELLE MEMORIAL INSTITUTE
00019   C  PACIFIC NORTHWEST LABORATORIES
00020   C  WATER & LAND RESOURCES DEPT.
00021   C
00022   C  AUTHOR(S):   DR FRIEDRICHS
00023   C                DW DAMSCHEN
00024   C
00025   C  DATE:  INITIAL VERSION  AUGUST  1979
00026   C         CURRENT VERSION  AUGUST  1979
00027   C
00028   C
00029         BYTE WLDES,WNAME,KEYBUF(80)
00030         INTEGER BKGRND,BKGLIN, WORKSP(35), PASSWD(4), STATUS(2), UIC(2)
00031         LOGICAL IN,LASTIN
00032   C
00033         DIMENSION H(512),T(512),INODE(280),ISZ(9),MSGBUF(20)
00034   C
00035         COMMON /DSP/MAIN(100),KEYOUT(100),BKGRND(100),BKGLIN(700),
00036        1         LBUT1(300),MESSY1(25),MESSY2(25),MESSX1(30),MESSX2(30),
00037        2         IGRAPH(5000),MESS2(35)
00038   C
00039         COMMON/HOR/ WLDES(12),IDUM(6),XC,YC,CASEL,IOBIT,ICBIT(2),
00040        1            IDM(236),NRETRN,MODE,WNAME(12),IOTYP,IOTYP,
00041        2            ICTYP,NVAR,MINX,MINY,MAXX,MAXY,IDM3(7)
00042   C
00043         EQUIVALENCE (WLDES(1),INODE(1)),(MSGBUF(1),WNAME(1))
00044   C
00045   C
00046         CALL MAINDF (MAIN,100,0,1,0,2)
00047         CALL CLKOFF
00048   C
00049         CALL OPFILE (3,'FRMTTMP',4000,,IPNT1)
00050   C
00051         SETUP-LIGHT-BUTTONS
00052   C
00053         IAG = 0
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101143  PAGE  00002
  00054          IRL = 0
  00055          IRR = 1
  00056    C
  00057          RETRIEVE-TEMPERATURE-DATA
  00058          CALCULATE-SCALE-FACTORS
  00059          DRAW-THE-BACKGROUND-CHART
  00060          DISPLAY-X-AND-Y-LINES-FOR-CHART
  00061          PLOT-DATA-POINTS-ON-GRAPH
  00062    C     WRITE (5,551) IRZ
  00063    551   FORMAT (' IRZ=',914)
  00064    C
  00065    C*************************************************************
  00066    C
  00067    C     START GRAPHIC CONTROL
  00068    C
  00069    C
  00070    C*************************************************************
  00071    C
  00072          IKR = 0
  00073    10    CALL GRATTN (1,ITYP,'LP')
  00074          IF(ITYP.NE.'LP') GO TO 10
  00075    C *** LIGHT PEN INTERRUPTS ***
  00076          CALL LTPEN (ID,ITIP)
  00077          IF(ITIP.EQ.1)
  00078         .  CONDITIONAL
  00079         .  .  (ID.EQ.1)
  00080         .  .  .  CALL STRTSB (MESSY1)
  00081         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00082         .  .  .  DECODE (NCHAR,219,KEYBUF) MINY
  00083    219  .  .  .  FORMAT (I4)
  00084         .  .  .  CALL STOPSB (MESSY1)
  00085         .  .  .  CALL STRTSB (MESSY2)
  00086         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00087         .  .  .  DECODE (NCHAR,219,KEYBUF) MAXY
  00088         .  .  .  CALL STOPSB (MESSY2)
  00089         .  .  .  YDIF = (MAXY-MINY)
  00090         .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00091         .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00092         .  .  ...FIN
  00093         .  .  (ID.EQ.2)
  00094         .  .  .  CALL STRTSB (MESSX1)
  00095         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00096         .  .  .  DECODE (NCHAR,219,KEYBUF) MINX
  00097         .  .  .  CALL STOPSB (MESSX1)
  00098         .  .  .  CALL STRTSB (MESSX2)
  00099         .  .  .  CALL GRECHO (KEYBUF,NCHAR,0,0,0,IKB)
  00100         .  .  .  DECODE (NCHAR,219,KEYBUF) MAXX
  00101         .  .  .  CALL STOPSB (MESSX2)
  00102         .  .  .  XDIF = (MAXX-MINX)+1
  00103         .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00104         .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00105         .  .  ...FIN
  00106         .  .  (ID.EQ.3)
  00107         .  .  .  MINX = MINXOR
  00108         .  .  .  MAXX = MAXXOR
  00109         .  .  .  MINY = MINYOR
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:01:43  PAGE  00003
  00110          .  .  .  MAXY = MAXYOR
  00111          .  .  .  YDIF = (MAXY-MINY)
  00112          .  .  .  XDIF = (MAXX-MINX)+1
  00113          .  .  .  DISPLAY-X-AND-Y-LINES-FOR-CHART
  00114          .  .  .  PLOT-DATA-POINTS-ON-GRAPH
  00115          .  .  ...FIN
  00116          .  .  (ID.EQ.4)
  00117          .  .  .  IOTYP = 1
  00118          .  .  .  ICON = 0
  00119          .  .  .  PASSWD(1) = 351
  00120          .  .  .  CALL NTINIT (STATUS,35,WORKSP)
  00121          .  .  .  CALL NTCONW (6,STATUS,ICON,'WLR45_','MNTR45',,UIC,1,PASSWD,TRAP)
  00122          .  .  .  CALL NTSNDW (6,STATUS,40,MSGBUF)
  00123          .  .  .  CONDITIONAL
  00124          .  .  .  .  (STATUS(1).LT.1)
  00125          .  .  .  .  .  WRITE(5,535) STATUS(1)
  00126    535   .  .  .  .  .  FORMAT(' TARGET IS NOT RECEPTIVE, STATUS =',I5)
  00127          .  .  .  .  .  CALL NTWAIT (6,STATUS)
  00128          .  .  .  .  ...FIN
  00129          .  .  .  .  (STATUS(1).EQ.1)
  00130          .  .  .  .  .  CALL NTDISW (6,STATUS,,)
  00131          .  .  .  .  ...FIN
  00132          .  .  .  .  (STATUS(1).GT.1)
  00133          .  .  .  .  .  WRITE(5,536) STATUS(1)
  00134    536   .  .  .  .  .  FORMAT(' ERROR DURING COMMUNICATION-+CODE=',I5)
  00135          .  .  .  .  ...FIN
  00136          .  .  ...FIN
  00137          .  .  .  CALL NTDISW (6,STATUS)
  00138          .  .  ...FIN
  00139          .  .  (ID.GE.20. AND .ID.LE.22)
  00140          .  .  .  NOCORD = 0
  00141          .  .  .  SELECT (ID)
  00142          .  .  .  .  (20)
  00143          .  .  .  .  .  NRETRN = 0
  00144          .  .  .  .  .  TASK = RAD50 ('CIRMIS')
  00145          .  .  .  .  ...FIN
  00146          .  .  .  .  (21)
  00147          .  .  .  .  .  NRETRN = 1
  00148          .  .  .  .  .  TASK = RAD50 ('WELSEL')
  00149          .  .  .  .  ...FIN
  00150          .  .  .  .  (22)
  00151          .  .  .  .  .  WHEN (XC.NE.0. AND .YC.NE.0)
  00152          .  .  .  .  .  .  NRETRN = 20
  00153          .  .  .  .  .  .  TASK = RAD50('MAPLOC')
  00154          .  .  .  .  .  ...FIN
  00155          .  .  .  .  .  ELSE
  00156          .  .  .  .  .  .  CALL STRTSB (MESS2)
  00157          .  .  .  .  .  .  CALL WAIT (2,2,MMM)
  00158          .  .  .  .  .  .  CALL STOPSB (MESS2)
  00159          .  .  .  .  .  .  NOCORD = 1
  00160          .  .  .  .  .  ...FIN
  00161          .  .  .  .  ...FIN
  00162          .  .  .  ...FIN
  00163          .  .  .  IF(NOCORD.EQ.0)
  00164          .  .  .  .  CALL DPW (IPNT0,0,,INODE,280,)
  00165          .  .  .  .  CALL REQUES (TASK,,IDS)
```

```
(FLECS VERSION 22.46)      14-SEP-79   08101143  PAGE  00004
  00166          .  .  .  .  III = 0
  00167          .  .  .  .  IF(III.EQ.0) CALL EXIT
  00168          .  .  .  ...FIN
  00169          .  .  ...FIN
  00170          .  ...FIN
  00171          ...FIN
  00172          GO TO 10

         ----------------------------------------

  00173          TO RETRIEVE-TEMPERATURE-DATA
  00174          .  CALL OPFILE (4,'WELSPEC',2,,IPNT0)
  00175          .  CALL OPR (IPNT0,0,,INODE,200,)
  00176    C     .
  00177    C  IDKIN--DISK NUMBER
  00178    C     .
  00179          .  IDKIN = INODE(50)
  00180          .  NPTSIN = INODE(51)
  00181          .  IDKEX  = INODE(52)
  00182          .  NPTSEX = INODE(53)
  00183          .  INALOC = INODE(54)
  00184    C     .
  00185    C  READ THE DATA IN THE INITIAL ALLOCATION
  00186    C     .
  00187          .  HADSI = IDKIN
  00188          .  WRDI  = NPTSIN*2,
  00189          .  TADSI = HADSI+INALOC
  00190          .  CALL OPR (IPNT1,HADSI,H,WRDI)
  00191          .  CALL OPR (IPNT1,TADSI,T,WRDI)
  00192    C     .
  00193    C  READ THE EXTENDED ALLOCATION (IF ANY)
  00194    C     .
  00195          .  IF (NPTSEX.NE.0)
  00196          .  .  HADSE = IDKEX
  00197          .  .  WRDE  = NPTSEX*2,
  00198    C     .  .
  00199    C  FOUR SECTORS ARE ALLOCATED FOR THE EXTENSION, TWO EACH FOR
  00200    C  TEMPERATURE AND TIME
  00201    C     .  .
  00202          .  .  TADSE = HADSE*2,
  00203          .  .  NDX   = NPTSIN+1
  00204          .  .  CALL OPR (IPNT1,HADSE,H(NDX),WRDE)
  00205          .  .  CALL OPR (IPNT1,TADSE,T(NDX),WRDE)
  00206          .  ...FIN
  00207          .  NPTS = NPTSIN + NPTSEX
  00208          ...FIN

         ----------------------------------------

  00209          TO CALCULATE-SCALE-FACTORS
  00210          .  IF (NVAR.EQ.0)
  00211          .  .  TMAX=0.0
  00212          .  .  TMIN=999999.
  00213          .  .  HMAX=0.0
  00214          .  .  HMIN=1000.
  00215          .  .  DO (I=1,NPTS)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:01:43  PAGE  00005
  00216          .  .  .  IF(T(I).GT.TMAX) TMAX=T(I)
  00217          .  .  .  IF(T(I).LT.TMIN) TMIN=T(I)
  00218          .  .  .  IF(H(I).GT.0.)
  00219          .  .  .  .  IF(H(I).GT.HMAX) HMAX=H(I)
  00220          .  .  .  .  IF(H(I).LT.HMIN) HMIN=H(I)
  00221          .  .  .  ...FIN
  00222          .  .  ...FIN
  00223          .  .  MINX=IFIX(TMIN/10000.)-1
  00224          .  .  MAXX=IFIX(TMAX/10000.)+1
  00225          .  .  MINY = IFIX(HMIN/10.)*10-10
  00226          .  .  MAXY = IFIX(HMAX/10.)*10+10
  00227          .  ...FIN
  00228          .  NVAR = 4
  00229          .  MINXOR = MINX
  00230          .  MAXXOR = MAXX
  00231          .  MINYOR = MINY
  00232          .  MAXYOR = MAXY
  00233          .  XDIF = (MAXX-MINX)+1
  00234          .  YDIF = (MAXY-MINY)
  00235          .  INCY = 10
  00236          .  IF (YDIF.LE.100.) INCY=5
  00237          .  IF (YDIF.LE.40.)  INCY=2
  00238          .  DSPX1 = 100.
  00239          .  DSPX2 = DSPX1 + 900.
  00240          .  DSPY1 = 100.
  00241          .  DSPY2 = DSPY1 + 800.
  00242          ...FIN
  00243    C

         ----------------------------------------

  00244          TO DRAW-THE-BACKGROUND-CHART
  00245          .  CALL SUBDF (BKGRND,100,IBG)
  00246          .  IBG = 2
  00247          .  CALL INTENS (5)
  00248          .  CALL APOS (IFIX(DSPX1),IFIX(DSPY1))
  00249          .  CALL BOX (IFIX(DSPX2-DSPX1),IFIX(DSPY2-DSPY1))
  00250          .  CALL CHRSCL (1)
  00251          .  CALL CHRTYP (0,1,0)
  00252          .  CALL APOS (30,325)
  00253          .  CALL TEXT ('  TEMPERATURE (DEGREES C)  ')
  00254          .  CALL CHRSCL (1)
  00255    1750  .  CALL APOS (460,60)
  00256          .  CALL CHRTYP (0,0,0)
  00257          .  CALL TEXT ('CALENDAR YEAR')
  00258    C     .
  00259          .  CALL CHRSCL (1)
  00260          .  CALL CHRTYP (0,0,0)
  00261          .  CALL APOS (415,950)
  00262          .  CALL TEXT ('TEMPERATURE HISTORY')
  00263          .  CALL CHRSCL (0)
  00264          .  CALL APOS (400,925)
  00265          .  CALL TEXT ('WELL DESIGNATION -- ')
  00266          .  CALL TEXT (WLDES,0,12)
  00267          .  CALL APOS (400,915)
  00268          .  CALL TEXT ('CASING ELEVATION -- ')
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:01:43  PAGE  00006
  00269          . CALL FNMBR (CASEL,'(F6.2)')
  00270          . CALL ENDSR(ISZ(1))
  00271          ...FIN

         ----------------------------------------

  00272          TO DISPLAY-X-AND-Y-LINES-FOR-CHART
  00273    C     .
  00274          . CALL SUBDF (BKGLIN,700,IBL)
  00275          . IBL = 2
  00276          . CALL CHRSCL (0)
  00277          . CALL INTENS (4)
  00278    C----DRAW Y LINES FOR CHART
  00279          . DSYINC = (DSPY2-DSPY1)/YDIF
  00280          . DO (JD = MINY,MAXY,INCY)
  00281          . . IYC = DSPY1 + FLOAT(JD-MINY)*DSYINC
  00282          . . CALL APOS (IFIX(DSPX1-30),IYC)
  00283          . . CALL INMBR (JD,'(I4)')
  00284          . . IF (JD.GT.MINY. AND .JD.LT.MAXY)
  00285          . . . CALL APOS (IFIX(DSPX1),IYC)
  00286          . . . CALL VEC  (IFIX(DSPX2),IYC)
  00287          . . ...FIN
  00288          . ...FIN
  00289    C----DRAW X LINES FOR CHART
  00290          . JDD = 1
  00291          . DSXINC = (DSPX2-DSPX1)/XDIF
  00292          . XADJ = DSXINC/2. + 7.
  00293          . DO (JD = MINX,MAXX+1)
  00294          . . X = DSPX1 + FLOAT(JD-MINX+1)*DSXINC
  00295          . . IF(JD.LE.MAXX)
  00296          . . . CALL APOS (IFIX(X-XADJ),IFIX(DSPY1-15))
  00297          . . . CALL INMBR (JD,'(I2)')
  00298          . . ...FIN
  00299          . . IF(JD.LE.MAXX)
  00300          . . . CALL APOS (IFIX(X),IFIX(DSPY1))
  00301          . . . SELECT (JDD)
  00302          . . . . (1)
  00303          . . . . . CALL RVEC (0,10)
  00304          . . . . . JDD = 2
  00305          . . . . ...FIN
  00306          . . . . (2)
  00307          . . . . . CALL RVEC (0,IFIX(DSPY2-DSPY1))
  00308          . . . . . JDD = 1
  00309          . . . . ...FIN
  00310          . . . ...FIN
  00311          . . ...FIN
  00312          . ...FIN
  00313          . CALL ENDSR(ISZ(2))
  00314          ...FIN

         ----------------------------------------

  00315          TO PLOT-DATA-POINTS-ON-GRAPH
  00316          . CALL SUBDF (IGRAPH,5000,I38)
  00317          . I38 = 2
  00318          . CALL INTENS (6)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101143  PAGE  00007
  00319          .  XSCALE = (DSPX2-DSPX1)/(XDIF*365.)
  00320          .  LASTIN = .FALSE.
  00321          .  DO (JK=1,NPTS)
  00322          .  .  IYR   = T(JK)/10000.
  00323          .  .  MO    = T(JK)/100.-FLOAT(IYR)*100.
  00324          .  .  IDA   = T(JK)-FLOAT(MO)*100.-FLOAT(IYR)*10000.
  00325    38    .  .  DAYS = (FLOAT(IYR-MINX)*365. + FLOAT(MO-1)*30.4 + FLOAT(IDA))
  00326          .  .  IX    = DAYS*XSCALE+DSPX1
  00327          .  .  X     = IX
  00328          .  .  IY = DSPY1 + (H(JK)-FLOAT(MINY))*DSYINC
  00329          .  .  Y   = IY
  00330    C  CHECK FOR POINT WITHIN REGION
  00331          .  .  IN = .FALSE.
  00332          .  .  IF(X.GE.DSPX1. AND .X.LE.DSPX2)
  00333          .  .  .  IF(Y.GE.DSPY1. AND .Y.LE.DSPY2)
  00334          .  .  .  .  WHEN (LASTIN) CALL VEC (IX,IY)
  00335          .  .  .  .  ELSE
  00336          .  .  .  .  .  CALL APOS (IX,IY)
  00337          .  .  .  .  ...FIN
  00338          .  .  .  .  LASTIN = .TRUE.
  00339          .  .  .  .  IN = .TRUE.
  00340          .  .  .  ...FIN
  00341          .  .  ...FIN
  00342          .  .  IF (.NOT.IN) LASTIN = .FALSE.
  00343          .  ...FIN
  00344          .  CALL ENDSB(ISZ(4))
  00345          .  CALL STRTSB (IGRAPH)
  00346          ...FIN

         ----------------------------------------

  00347          TO SETUP-LIGHT-BUTTONS
  00348    C     .
  00349          .  CALL SUBOF (LBUT1,300,0)
  00350          .  CALL AREA (1)
  00351          .  CALL CHRSCL (0)
  00352          .  CALL INTENS (4)
  00353          .  CALL APOS (0,700)
  00354          .  CALL TEXT ('  LIGHT PEN OPTIONS')
  00355          .  CALL INTENS (7)
  00356          .  CALL APOS (0,685)
  00357          .  CALL BOX (150,40)
  00358          .  CALL APOS (5,690)
  00359          .  CALL BOX (140,30)
  00360          .  CALL APOS (30,625)
  00361          .  CALL TEXT ('SCALE CHANGE',-3)
  00362          .  CALL RVEC (84,0)
  00363          .  CALL MENU (590,30,1,'TEMPERATURE SCALE','TIME SCALE',
  00364        1.  'ORIGINAL SCALE')
  00365          .  CALL AREA (1)
  00366          .  CALL APOS (30,450)
  00367          .  CALL TEXT ('OUTPUT DEVICES',-3)
  00368          .  CALL RVEC (90,0)
  00369          .  CALL MENU (415,30,4,'CAL-COMP',
  00370        1.  'GOULD','TEKTRONIX')
  00371          .  CALL AREA (1)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08101143  PAGE  00008
  00372           .  CALL APOS (30,150)
  00373           .  CALL TEXT ('PROGRAM CONTROLS',-3)
  00374           .  CALL RVEC (110,0)
  00375           .  CALL MENU (115,30,20,'RETURN TO CIRMIS CONTROL',
  00376         1.  'RETURN TO WELL SELECT','DISPLAY MAP LOCATION')
  00377           .  CALL AREA (0)
  00378           .  CALL ENDSR(ISZ(5))
  00379    C      .
  00380    C *** OUTPUT MESSAGES
  00381           .  CALL SUBDF (MESSY1,25,1)
  00382           .  CALL CHRSCL (0)
  00383           .  CALL INTENS (7)
  00384           .  CALL BLINK (1)
  00385           .  CALL APOS (0,30)
  00386           .  CALL TEXT ('ENTER MINIMUM TEMPERATURE (I4)')
  00387           .  CALL BLINK (0)
  00388           .  CALL ENDSR(ISZ(6))
  00389    C      .
  00390           .  CALL SUBDF (MESSY2,25,1)
  00391           .  CALL CHRSCL (0)
  00392           .  CALL INTENS (7)
  00393           .  CALL BLINK (1)
  00394           .  CALL APOS (0,50)
  00395           .  CALL TEXT ('ENTER MAXIMUM TEMPERATURE (I4)')
  00396           .  CALL BLINK (0)
  00397           .  CALL ENDSR (ISZ(7))
  00398    C      .
  00399           .  CALL SUBDF (MESSX1,30,1)
  00400           .  CALL CHRSCL (0)
  00401           .  CALL INTENS (7)
  00402           .  CALL BLINK (1)
  00403           .  CALL APOS (0,30)
  00404           .  CALL TEXT ('ENTER MINIMUM TIME IN YEARS (I4)')
  00405           .  CALL BLINK (0)
  00406           .  CALL ENDSR(ISZ(8))
  00407    C      .
  00408           .  CALL SUBDF (MESSX2,30,1)
  00409           .  CALL CHRSCL (0)
  00410           .  CALL INTENS (7)
  00411           .  CALL BLINK (1)
  00412           .  CALL APOS (0,50)
  00413           .  CALL TEXT ('ENTER MAXIMUM TIME IN YEARS (I4)')
  00414           .  CALL BLINK (0)
  00415           .  CALL ENDSR(ISZ(9))
  00416    C      .
  00417           .  CALL SUBDF (MESS2,35,1)
  00418           .  CALL CHRSCL (0)
  00419           .  CALL AREA (1)
  00420           .  CALL BLINK (1)
  00421           .  CALL APOS (0,200)
  00422           .  CALL TEXT ('COORDINATES NOT AVAILABLE FOR THIS WELL')
  00423           .  CALL BLINK (0)
  00424           .  CALL AREA (0)
  00425           .  CALL ENDSR
  00426          ...FIN
  00427         END
```

```
        ----------------------------------------

              PROCEDURE CROSS-REFERENCE TABLE

        00209 CALCULATE-SCALE-FACTORS
              00058

        00272 DISPLAY-X-AND-Y-LINES-FOR-CHART
              00060  00090  00103  00113

        00244 DRAW-THE-BACKGROUND-CHART
              00059

        00315 PLOT-DATA-POINTS-ON-GRAPH
              00061  00091  00104  00114

        00173 RETRIEVE-TEMPERATURE-DATA
              00057

        00347 SETUP-LIGHT-BUTTONS
              00051


        (FLECS VERSION 22.46)

        ----------------------------------------

00428           SUBROUTINE GRECHO(KEYBUF,NCHAR,IX,IY,ICS,IKB)
00429   C
00430           BYTE KEYBUF(80)
00431           INTEGER DEL,CR
00432           COMMON/DSP/ MAIN(100),KEYOUT(100)
00433   C
00434           DEL = "177
00435           CR = "15
00436           CALL SUBOF(KEYOUT,100,IKB)
00437           IKB = 2
00438           CALL INTENS (4)
00439           CALL CHRSCL (ICS)
00440           CALL APOS (IX,IY)
00441           CALL TEXT ('>SE')
00442           CALL ENDSB
00443   C
00444           NCHAR = 0
00445   C
00446   10      CALL GRATIN (1,ITYP,'KB')
00447           IF (ITYP.NE.'KB') GO TO 10
00448           CALL KEYBRD (KEY)
00449           IF (KEY.EQ.CR) GO TO 30
00450           IF (NCHAR.EQ.80) GO TO 30
00451           IF (NCHAR.LT.0) GO TO 10
00452           NCHAR = NCHAR+1
00453           KEYBUF(NCHAR) = KEY
00454           IF (KEY.EQ.DEL)
00455          .  NCHAR = NCHAR-2
```

```
(FLECS VERSION 22.46)      14-SEP-79   00101143  PAGE  00010
  00456           ...FIN
  00457           CALL SUBDF(KEYOUT,100,IKB)
  00458           CALL INTENS (4)
  00459           CALL CHRSCL (IC8)
  00460           CALL APOS(IX,IY)
  00461           CALL TEXT (KEYBUF,0,NCHAR)
  00462           CALL TEXT ('>SE')
  00463           CALL ENDSB
  00464           GO TO 10
  00465    30     RETURN
  00466           END

           (FLECS VERSION 22.46)

           ------------------------------------
```

DSPSEL.FLX

WELL DISPLAY SELECTOR

```
        ----------------------------------------

C ******************** [351,101]USPSEL.FLX ********************
C
C
C  PROGRAM ID:  XX
C
C  PROGRAM DESCRIPTION:
C        PROGRAM TO MAKE WELL SELECTIONS FROM A DISPLAYED
C        BACKGROUND MAP AND DISPLAYED WELL LOCATIONS.
C
C  DATA FILES:
C       NAME                      LUN            TYPE           ACCESS
C       VS60                      1              --             ---
C       FILEQ(DB0)---WELLHDR      3              RAN             R
C       FILEQ(DK1)---WELSPEC      4              RAN            R/W
C       FILEQ(DB0)---WELL199      3              RAN             R
C           (ALSO--WELL299E,WELL299W,WELL399,WELL699,WELL1199,WELL3099)
C       DI:---MAP DISPLAYS        6              BIN             R
C             (EX:  AREA01.MAP AND WLL199.DSP)
C
C
C
C
C  BATTELLE MEMORIAL INSTITUTE
C  PACIFIC NORTHWEST LABORATORIES
C  WATER & LAND RESOURCES DEPT.
C
C  AUTHOR(S):    DR FRIEDRICHS
C
C  DATE:  INITIAL VERSION  FEBRUARY 27, 1978
C         CURRENT VERSION  MARCH 5, 1979
C
C
      BYTE WLDES, LAST, WNAME(12)
      BYTE BUFF(512)
      INTEGER WELLS
      REAL MAXMAP
      LOGICAL START
C
      DIMENSION KBUF(256), IBUF(280)
C
      COMMON /DSP/ MAIN(100),WELLS(4360),MAP(3000), IWELL(200),
     1             LBUT(300), ISPECS(100)
C
      COMMON/HDR/ WLDES(12),IDUM(6),XC,YC,CASEL,IDBIT,ICBIT(2),
     1            IDM(234),IWNAM,IWHDR,NRETRN,MODE,IDM2(21)
C
      EQUIVALENCE (IBUF(1),WLDES(1))
      EQUIVALENCE (KBUF(1),BUFF(1))
C
C
C
C
      CALL OPFILE (4,'WELSPEC',2,,IPNT0)
```

```
(FLECS VERSION 22.06)     14-SEP-79   08102123  PAGE  00002
  00054          CALL DPR (IPNT0,0,,IBUF,200,)
  00055          CALL DPFILE (3,'WELLNAM',101,,IWNAM)
  00056          CALL DPFILE (3,'WELLHDR',6000,,IWHDR)
  00057          TNL = 0
  00058          START = .TRUE.
  00059    C
  00060          CALL MAINDF (MAIN,100,0,1)
  00061          CALL CLKOFF
  00062    C
  00063          SETUP-DISPLAY-FILES
  00064    C
  00065    C
  00066    10    CALL GRATTN (1,ITYP,'LP')
  00067          IF (ITYP.NE.'LP') GO TO 10
  00068          CALL LIPEN (ID,ITIP)
  00069          IF (ITIP.EQ.1)
  00070    C     .
  00071          .  CONDITIONAL
  00072    C  *** SELECT REGION
  00073          .  .  (ID.EQ.1)
  00074          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA01.MAP',MAP)
  00075          .  .  .  IF (START) CALL CALLSB (MAP)
  00076          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL199.DSP',WELLS)
  00077          .  .  .  IF (START) CALL CALLSB (WELLS)
  00078          .  .  .  START = .FALSE.
  00079          .  .  .  CALL DPFILE (3,'WELL199',14,,IDK)
  00080          .  .  .  XINC = 1000.
  00081          .  .  .  YINC = 1000.
  00082          .  .  .  NODESX = 16
  00083          .  .  .  NODESY = 16
  00084          .  .  .  XADJ = 50000.
  00085          .  .  .  YADJ = 102000.
  00086          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00087          .  .  ...FIN
  00088          .  .  (ID.EQ.2)
  00089          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA2E.MAP',MAP)
  00090          .  .  .  IF (START) CALL CALLSB (MAP)
  00091          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL299E.DSP',WELLS)
  00092          .  .  .  IF (START) CALL CALLSB (WELLS)
  00093          .  .  .  START = .FALSE.
  00094          .  .  .  CALL DPFILE (3,'WELL299E',14,,IDK)
  00095          .  .  .  XINC = 1000.
  00096          .  .  .  YINC = 1000.
  00097          .  .  .  NODESX = 16
  00098          .  .  .  NODESY = 16
  00099          .  .  .  XADJ = 50000.
  00100          .  .  .  YADJ = 102000.
  00101          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00102          .  .  ...FIN
  00103          .  .  (ID.EQ.3)
  00104          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA2W.MAP',MAP)
  00105          .  .  .  IF (START) CALL CALLSB (MAP)
  00106          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL299W.DSP',WELLS)
  00107          .  .  .  IF (START) CALL CALLSB (WELLS)
  00108          .  .  .  START = .FALSE.
  00109          .  .  .  CALL DPFILE (3,'WELL299W',20,,IDK)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102123  PAGE  00003
  00110          .  .  .  XINC = 1000.
  00111          .  .  .  YINC = 1000.
  00112          .  .  .  NODESX = 19
  00113          .  .  .  NODESY = 20
  00114          .  .  .  XADJ = 25000.
  00115          .  .  .  YADJ = 99000.
  00116          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00117          .  .  ...FIN
  00118          .  .  (ID.EQ.4)
  00119          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA03.MAP',MAP)
  00120          .  .  .  IF (START) CALL CALLSB (MAP)
  00121          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL399.DSP',WELLS)
  00122          .  .  .  IF (START) CALL CALLSB (WELLS)
  00123          .  .  .  START = .FALSE.
  00124          .  .  .  CALL DPFILE (3,'WELL399',3.,IDK)
  00125          .  .  .  XINC = 500.
  00126          .  .  .  YINC = 500.
  00127          .  .  .  NODESX = 17
  00128          .  .  .  NODESY = 26
  00129          .  .  .  XADJ = 118000.
  00130          .  .  .  YADJ = 38000.
  00131          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00132          .  .  ...FIN
  00133          .  .  (ID.EQ.5)
  00134          .  .  .  CALL RESSDF (6,'DI:[351,200]HANFORD.MAP',MAP)
  00135          .  .  .  IF (START) CALL CALLSB (MAP)
  00136          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL699.DSP',WELLS)
  00137          .  .  .  IF (START) CALL CALLSB (WELLS)
  00138          .  .  .  START = .FALSE.
  00139          .  .  .  CALL DPFILE (3,'WELL699',19.,IDK)
  00140          .  .  .  XINC = 1000.
  00141          .  .  .  YINC = 1000.
  00142          .  .  .  NODESX = 135
  00143          .  .  .  NODESY = 173
  00144          .  .  .  XADJ = 0.
  00145          .  .  .  YADJ = 0.
  00146          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00147          .  .  ...FIN
  00148          .  .  (ID.EQ.6)
  00149          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA11.MAP',MAP)
  00150          .  .  .  IF (START) CALL CALLSB (MAP)
  00151          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL1199.DSP',WELLS)
  00152          .  .  .  IF (START) CALL CALLSB (WELLS)
  00153          .  .  .  START = .FALSE.
  00154          .  .  .  CALL DPFILE (3,'WELL1199',14.,IDK)
  00155          .  .  .  XINC = 1000.
  00156          .  .  .  YINC = 1000.
  00157          .  .  .  NODESX = 16
  00158          .  .  .  NODESY = 16
  00159          .  .  .  XADJ = 50000.
  00160          .  .  .  YADJ = 102000.
  00161          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00162          .  .  ...FIN
  00163          .  .  (ID.EQ.7)
  00164          .  .  .  CALL RESSDF (6,'DI:[351,200]AREA30.MAP',MAP)
  00165          .  .  .  IF (START) CALL CALLSB (MAP)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:02:23  PAGE  00004
  00166          .  .  .  CALL RESSDF (6,'DI:[351,200]WLL3099.DSP',WELLS)
  00167          .  .  .  IF (START) CALL CALLSB (WELLS)
  00168          .  .  .  START = .FALSE.
  00169          .  .  .  CALL OPFILE (3,'WELL3099',14,,IDK)
  00170          .  .  .  XINC = 1000.
  00171          .  .  .  YINC = 1000.
  00172          .  .  .  NODESX = 16
  00173          .  .  .  NODESY = 16
  00174          .  .  .  XADJ = 50000.
  00175          .  .  .  YADJ = 102000.
  00176          .  .  .  CALCULATE-DISPLAY-SCALE-FACTOR
  00177          .  .  ...FIN
  00178    C  *** RETURN TO DATA SELECT PAGE
  00179          .  .  (ID.EQ.20)
  00180          .  .  .  NRETRN = 1
  00181          .  .  .  CALL OPW (IPNT0,0,,IBUF,280,)
  00182          .  .  .  TASK = RAD50 ('WELSEL')
  00183          .  .  .  CALL REQUES (TASK,,IDS)
  00184          .  .  .  IF (ID.EQ.20) CALL EXIT
  00185          .  .  ...FIN
  00186          .  .  (ID.EQ.21)
  00187          .  .  .  IF (ID.EQ.21) STOP
  00188          .  .  ...FIN
  00189    C  *** LP HIT ON WELL DISPLAY
  00190          .  .  (ID.GE.101. AND.ID.LE.900)
  00191          .  .  .  RETRIEVE-NAME-FROM-BUFFER
  00192          .  .  .  CONVERT-TO-MAP-COORDINATES-AND-DISPLAY
  00193          .  .  ...FIN
  00194          .  ...FIN
  00195          ...FIN
  00196    C
  00197          GO TO 10

         ----------------------------------------

  00198          TO CALCULATE-DISPLAY-SCALE-FACTOR
  00199          .  XMAX = FLOAT(NODESX-1) * XINC
  00200          .  YMAX = FLOAT(NODESY-1) * YINC
  00201          .  MAXMAP = AMAX1(XMAX,YMAX)
  00202    C  CALCULATE "DISPLAY" SCALE FACTOR
  00203          .  WHEN(YMAX.GT.XMAX) DSMAX=900.
  00204          .  ELSE
  00205          .  .  TEST = 900./1023.
  00206          .  .  TMAP = YMAX/XMAX
  00207          .  .  WHEN (TMAP.LE.TEST) DSMAX = 1023.
  00208          .  .  ELSE
  00209          .  .  .  DSMAX = (TEST/TMAP)*1023.
  00210          .  .  ...FIN
  00211          .  ...FIN
  00212          .  DSCALE = DSMAX/MAXMAP
  00213          ...FIN

         ----------------------------------------

  00214          TO RETRIEVE-NAME-FROM-BUFFER
  00215          .  NADR = (ID-101)/42
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102123  PAGE  00005
  00216          .  ADR = NADR
  00217          .  NWELL = ID-100-NADR*42
  00218          .  CALL DPR (IDK,ADR,KBUF,256,)
  00219          .  III = (NWELL-1)*12+1
  00220          .  DO (J=1,12)
  00221          .  .  WNAME(J) = BUFF(III)
  00222          .  .  III = III+1
  00223          .  ...FIN
  00224          .  CALL RTVHDR(WNAME,FDADR)
  00225          ...FIN

         ----------------------------------------

  00226          TO CONVERT-TO-MAP-COORDINATES-AND-DISPLAY
  00227          .  PLANTX = XC
  00228          .  PLANTY = YC
  00229          .  YCOORD = PLANTY - YADJ
  00230    C     .
  00231          .  XCOORD = PLANTX - XADJ
  00232          .  IF (XC.EQ.0., AND .YC.EQ.0.)
  00233          .  .  XCOORD = 0.
  00234          .  .  YCOORD = 0.
  00235          .  ...FIN
  00236          .  IX = XCOORD*DSCALE
  00237          .  IY = YCOORD*DSCALE
  00238          .  CALL SUBDF (IWELL,200,IWL)
  00239          .  IWL = 2
  00240          .  CALL INTENS (7)
  00241          .  CALL CHRSCL (1)
  00242          .  CALL APOS (IX,IY)
  00243          .  CALL VEC (512,910)
  00244          .  CALL APOS (420,950)
  00245          .  CALL TEXT ('SELECTED WELL')
  00246          .  CALL APOS (405,910)
  00247          .  CALL BOX (210,60)
  00248          .  CALL APOS (434,925)
  00249          .  CALL TEXT (WLDES,0,12)
  00250          .  CALL AREA (1)
  00251          .  CALL INTENS (4)
  00252          .  CALL CHRSCL (0)
  00253          .  CALL APOS (110,80)
  00254          .  CALL TEXT (WLDES,-15,12)
  00255          .  CALL FNMBR (CASEL,'(F7.2)',-15)
  00256          .  CALL FNMBR (XC,'(F7.0)',-15)
  00257          .  CALL FNMBR (YC,'(F7.0)',-15)
  00258          .  CALL AREA (0)
  00259          .  CALL ENDSB
  00260          ...FIN

         ----------------------------------------

  00261          TO SETUP-DISPLAY-FILES
  00262          .  CALL SUBDF (LBUT,300)
  00263          .  CALL AREA (1)
  00264          .  CALL INTENS (7)
  00265          .  CALL CHRSCL (0)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102123  PAGE  00006
  00266          .  CALL APOS (0,800)
  00267          .  CALL TEXT ('  SELECT DESIRED WELL CATEGORY')
  00268          .  CALL APOS (0,785)
  00269          .  CALL BOX (210,40)
  00270          .  CALL INTENS (2)
  00271          .  CALL APOS (0,790)
  00272          .  CALL BOX (200,30)
  00273          .  CALL INTENS (7)
  00274          .  CALL MENU (760,30,1,'199 AREA WELLS','299 EAST AREA WELLS',
  00275        1.  '299 WEST AREA WELLS','399 AREA WELLS','699 AREA WELLS',
  00276        2.  '1199 AREA WELLS','3099 AREA WELLS')
  00277          .  CALL MENU (400,30,20,'RETURN TO DATA SELECT','STOP')
  00278          .  CALL AREA (0)
  00279          .  CALL ENDSB
  00280    C     .
  00281          .  CALL SUBDF (ISPECS,100)
  00282          .  CALL AREA (1)
  00283          .  CALL INTENS (4)
  00284          .  CALL CHRSCL (0)
  00285          .  CALL APOS (0,100)
  00286          .  CALL TEXT ('CURRENTLY SELECTED WELL DATA:',-3)
  00287          .  CALL RVEC (200,0)
  00288          .  CALL APOS (0,80)
  00289          .  CALL TEXT ('DESIGNATION  =',-15)
  00290          .  CALL TEXT ('CASING ELEV  =',-15)
  00291          .  CALL TEXT ('X COORDINATE =',-15)
  00292          .  CALL TEXT ('Y COORDINATE =',-15)
  00293          .  CALL AREA (0)
  00294          .  CALL ENDSB
  00295          ...FIN
  00296          END

          ----------------------------------------

                 PROCEDURE CROSS-REFERENCE TABLE

          00198 CALCULATE-DISPLAY-SCALE-FACTOR
                 00086  00101  00116  00131  00146  00161  00176

          00226 CONVERT-TO-MAP-COORDINATES-AND-DISPLAY
                 00192

          00214 RETRIEVE-NAME-FROM-BUFFER
                 00191

          00261 SETUP-DISPLAY-FILES
                 00063


          (FLECS VERSION 22.46)

          ----------------------------------------
```

APPLIC.FLX

SPECIAL APPLICATIONS

```
(FLECS VERSION 22.46)     14-SEP-79   00102:53  PAGE  00001

      ----------------------------------------

00001   C ********************  (351,101]APPLIC.FLX  ******************
00002   C
00003   C  MONITOR CONTROL PROGRAM FOR CIRMIS
00004   C
00005   C
00006   C  D.R. FRIEDRICHS
00007   C
00008   C
00009         BYTE FNAME,SRCMAP,DSPMAP,NONE(4)
00010         BYTE KEYBUF(80)
00011         LOGICAL NEWFIL
00012         REAL MAXMAP
00013   C
00014         DIMENSION NDIGIT(78),SVAL(256)
00015   C
00016         COMMON/DSP/ MAIN(100),KEYOUT(100),ISPECS(300),LABELS(500),
00017        1            ISTAR(20)
00018   C
00019         COMMON/IND/ INITDZ,DZXMAX,DZYMAX,DZMAX,NODESX,NODESY,XINC,YINC,
00020        1            NSX,TOTSEC,TOTX,FNAME(30),SRCMAP(30),DSPMAP(30),
00021        2            XMAX,YMAX,MAXMAP,DSMAX,IARG,WRKMIN,WRKMAX,LINE
00022   C
00023         EQUIVALENCE (NDIGIT(1),INITDZ)
00024   C
00025         DATA NONE /'N','O','N','E'/
00026   C
00027   C
00028         CALL MAINOF(MAIN,200,0,1,0,2)
00029         CALL CLKOFF
00030         IMN = 0
00031         IKB = 0
00032         IST = 0
00033         NEWFIL = .FALSE.
00034         INPUT = .FALSE.
00035         CALL ERRSET (29,,TRUE.,,FALSE.,,TRUE.,,FALSE.,15)
00036   C
00037         SET-UP-DISPLAY-FILES
00038   C
00039         CALL OPIND (4,'SURFSPC',ISTAT,SIZE)
00040         CONDITIONAL
00041        .  (ISTAT.EQ.-1)
00042   C  FILE DOESNT EXIST-SET UP NEW ONE
00043        .  .  INITDZ = 0
00044        .  .  NODESX = 0
00045        .  .  NODESY = 0
00046        .  .  XINC  = 0.
00047        .  .  YINC  = 0.
00048        .  .  IARG = 11
00049        .  .  WRKMIN = 0.
00050        .  .  WRKMAX = 0.
00051        .  .  DO (J=1,4) FNAME(J) = NONE(J)
00052        .  .  DO (J=5,30) FNAME(J) = ' '
00053        .  .  DO (J=1,4) SRCMAP(J) = NONE(J)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102153  PAGE  00002
  00054          .  .  DO (J=5,30) SRCMAP(J) = ' '
  00055          .  .  NEWFIL = .TRUE.
  00056          .  .  CALL OPFILE (4,'SURFSPC',1,,IPNT0)
  00057          .  .  CALL OPW(IPNT0,0.,,NDIGIT,78.)
  00058          .  ...FIN
  00059          .  (ISTAT.EQ.0)
  00060    C  FILE EXISTS AND IS NOT OPEN
  00061          .  .  CALL OPFILE (4,'SURFSPC',1,,IPNT0)
  00062          .  .  CALL OPR (IPNT0,0.,,NDIGIT,78.)
  00063          .  .  INITDZ = 0
  00064          .  .  IF (SRCMAP(1).NE.'N') INITDZ = 1
  00065          .  ...FIN
  00066          .  (ISTAT.GT.0)
  00067    C  FILE EXISTS BUT IS OPEN
  00068          .  .  WRITE(5,510)
  00069    510   .  .  FORMAT(' FILE 0 SPECIFICATION FILE IS OPEN')
  00070          .  .  III = 0
  00071          .  .  IF (III.EQ.0) STOP
  00072          .  ...FIN
  00073          ...FIN
  00074    C
  00075    C
  00076    C
  00077          ICHAR = ICHR(SRCMAP)
  00078          KCHAR = ICHR(FNAME)
  00079          DISPLAY-FILE-SPECIFICATIONS
  00080    C
  00081    C
  00082    10    CALL GRATTN (1,ITYP,'LP')
  00083    C
  00084          IF (ITYP.NE.'LP') GO TO 10
  00085          CALL LTPEN (ID,ITIP)
  00086          IF (ITIP.EQ.1)
  00087          .  CONDITIONAL
  00088          .  .  (ID.GE.1. AND .ID.LE.7)
  00089          .  .  .  DISPLAY-STAR
  00090          .  .  .  IF (NEWFIL)
  00091          .  .  .  .  NSX = (NODESX+127)/128
  00092          .  .  .  .  TOTSEC = NSX*NODESY
  00093          .  .  .  .  TOTX = NODESX*2
  00094          .  .  .  .  XMAX = FLOAT(NODESX-1) * XINC
  00095          .  .  .  .  YMAX = FLOAT(NODESY-1) * YINC
  00096          .  .  .  .  MAXMAP = AMAX1(XMAX,YMAX)
  00097    C  CALCULATE "DISPLAY" SCALE FACTOR
  00098          .  .  .  .  WHEN(YMAX.GT.XMAX) DSMAX=900.
  00099          .  .  .  .  ELSE
  00100          .  .  .  .  .  TEST = 900./1023.
  00101          .  .  .  .  .  TMAP = YMAX/XMAX
  00102          .  .  .  .  .  WHEN (TMAP.LE.TEST) DSMAX = 1023.
  00103          .  .  .  .  .  ELSE
  00104          .  .  .  .  .  .  DSMAX = (TEST/TMAP)*1023.
  00105          .  .  .  .  .  ...FIN
  00106          .  .  .  .  ...FIN
  00107          .  .  .  .  DETERMINE-MAP-DISPLAY-FILE-NAME
  00108          .  .  .  .  CALL OPW (IPNT0,0.,,NDIGIT,78.)
  00109          .  .  .  .  CALL OPFILE (4,'SURFTMP',TOTSEC,IPNT1)
```

```
                . . . .  OPEN (UNIT=7,NAME=FNAME,TYPE='OLD',FORM='UNFORMATTED',ERR=20)
      C   BINARY FILE EXISTS--WRITE TO FILE Q
                . . . .  DO(J=1,NODESY)
                . . . . .  ADR = (J-1)*NSX
                . . . . .  READ (7) (SVAL(I),I=1,NODESX)
                . . . . .  CALL DPW (IPNT1,ADR,SVAL,TOTX)
                . . . . .  ...FIN
                . . . .  CLOSE (UNIT=7)
                . . . .  GO TO 25
      C   BINARY FILE DOES NOT EXIST--ZERO FILE Q
      20        . . . .  DO (J=1,NODESX) SVAL(J)=0.0
                . . . .  DO (J=1,NODESY)
                . . . . .  ADR = (J-1)*NSX
                . . . . .  CALL DPW(IPNT1,ADR,SVAL,TOTX)
                . . . .  ...FIN
      C         . . . .
      25        . . . .  CONTINUE
                . . .  ...FIN
                . . .  IF (ID.EQ.1) TASK = RAD50('MAPDRW')
                . . .  IF (ID.EQ.2) TASK = RAD50('SURFGN')
                . . .  IF (ID.EQ.3) TASK = RAD50('CONTUR')
                . . .  IF (ID.EQ.4) TASK = RAD50('MXEDIT')
                . . .  IF (ID.EQ.5) TASK = RAD50('FINDZ')
                . . .  IF (ID.EQ.6) TASK = RAD50('FILLZ')
                . . .  IF (ID.EQ.7) TASK = RAD50('CIRMIS')
                . . .  CALL REQUES (TASK,,IDS)
                . . .  IF(IDS.GT.0) CALL EXIT
                . . .  WRITE(5,530) IDS
      530       . . .  FORMAT(' REQUEST ERROR.  IDS = ',I5)
                . . .  III = 0
                . . .  IF (III.EQ.0) STOP
                . .  ...FIN
                . .  (ID.GE.21. AND .ID.LE.26)
                . . .  DO (J=1,30) KEYBUF(J) = ' '
                . . .  CALL GRECHO (KEYBUF,NCHAR,628,170,0,0,IKB)
                . . .  SELECT (ID)
                . . . .  (21) DECODE(NCHAR,211,KEYBUF) NODESX
                . . . .  (22) DECODE(NCHAR,211,KEYBUF) NODESY
                . . . .  (23) DECODE(NCHAR,212,KEYBUF) XINC
                . . . .  (24) DECODE(NCHAR,212,KEYBUF) YINC
                . . . .  (25)
                . . . . .  DECODE(NCHAR,213,KEYBUF) SRCMAP
                . . . . .  ICHAR = ICHR(SRCMAP)
                . . . .  ...FIN
                . . . .  (26)
                . . . . .  DECODE(NCHAR,213,KEYBUF) FNAME
                . . . . .  KCHAR = ICHR(FNAME)
                . . . .  ...FIN
                . . .  ...FIN
      211       . . .  FORMAT(I5)
      212       . . .  FORMAT(F10.0)
      213       . . .  FORMAT(30A1)
                . . .  DISPLAY-FILE-SPECIFICATIONS
                . . .  NEWFIL = .TRUE.
                . . .  IF (ID.GE.21. AND .ID.LE.25) INITDZ = 0
                . .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102153  PAGE  00004
  00166          .  ...FIN
  00167          ...FIN
  00168          GO TO 10

         ----------------------------------------

  00169          TO DISPLAY-STAR
  00170          .  CALL SUBDF (ISTAR,20,IST)
  00171          .  IST = 2
  00172          .  CALL INTENS (4)
  00173          .  CALL CHRSCL (2)
  00174          .  IY = 820-(ID-1)*40
  00175          .  CALL APOS (330,IY)
  00176          .  CALL TEXT ('*')
  00177          .  CALL ENDSR
  00178          .  CALL WAIT (1,2,MMM)
  00179          ...FIN

         ----------------------------------------

  00180          TO DETERMINE-MAP-DISPLAY-FILE-NAME
  00181          .  I = 0
  00182          .  DO (J=1,ICHAR)
  00183          .  .  I = I+1
  00184          .  .  DSPMAP(I) = SRCMAP(J)
  00185          .  .  IF (SRCMAP(J).EQ.'.') GO TO 18
  00186          .  ...FIN
  00187          .  DO (J=1,4) SRCMAP(J)=NONE(J)
  00188          .  GO TO 19
  00189    18    .  DSPMAP(I+1) = 'M'
  00190          .  DSPMAP(I+2) = 'A'
  00191          .  DSPMAP(I+3) = 'P'
  00192          .  ICHAR = ICHR(DSPMAP)
  00193    19    .  CONTINUE
  00194          ...FIN

         ----------------------------------------

  00195          TO DISPLAY-FILE-SPECIFICATIONS
  00196          .  CALL SUBDF (I8PEC8,300,IMN)
  00197          .  IMN = 2
  00198          .  CALL CHRSCL(1)
  00199          .  CALL INTENS(4)
  00200          .  IY = 450
  00201          .  DO (J=1,6)
  00202          .  .  IX = 610
  00203          .  .  IF(J.LE.2) IX = 652
  00204          .  .  CALL APOS (IX,IY)
  00205    C     .  .  CALL POINTR (J)
  00206          .  .  CALL INTENS (4)
  00207          .  .  SELECT (J)
  00208          .  .  .  (1) CALL INMBR (NODESX,'(I5)')
  00209          .  .  .  (2) CALL INMBR (NODESY,'(I5)')
  00210          .  .  .  (3) CALL FNMBR (XINC,'(FB.1)')
  00211          .  .  .  (4) CALL FNMBR (YINC,'(FB.1)')
  00212          .  .  .  (5) CALL TEXT (SRCMAP,0,ICHAR)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08102153  PAGE  00005
  00213          .  .  .  (6) CALL TEXT (FNAME,0,KCHAR)
  00214          .  .  ...FIN
  00215          .  .  IY = IY-40
  00216          .  ...FIN
  00217          .  CALL ENDSR
  00218          ...FIN

         ----------------------------------------

  00219          TO SET-UP-DISPLAY-FILES
  00220    C     .
  00221          .  CALL SUBDF (LABELS,500,0)
  00222          .  CALL CHRSCL (2)
  00223          .  CALL INTENS (7)
  00224          .  CALL CHRTYP (1,0,0)
  00225          .  CALL APOS (145,950)
  00226          .  CALL TEXT ('SPECIAL APPLICATIONS SELECTION ROUTINE')
  00227          .  CALL APOS (100,915)
  00228          .  CALL BOX (890,90)
  00229          .  CALL INTENS (2)
  00230          .  CALL RPOS (5,5)
  00231          .  CALL BOX (880,80)
  00232    C     .
  00233          .  CALL LPHIT (1)
  00234          .  IX = 330
  00235          .  IY = 860
  00236          .  CALL INTENS (5)
  00237          .  CALL CHRSCL (1)
  00238          .  CALL APOS (IX,IY)
  00239          .  CALL TEXT ('SELECT A SPECIAL FUNCTION:',-3)
  00240          .  CALL RVEC (364,0)
  00241          .  IX = 370
  00242          .  IY = IY-40
  00243          .  DO (J=1,7,1)
  00244          .  .  CALL NAME (J)
  00245          .  .  CALL APOS (IX,IY)
  00246          .  .  SELECT (J)
  00247          .  .  .  (1) CALL TEXT ('MAP GENERATOR')
  00248          .  .  .  (2) CALL TEXT ('SURFACE GENERATOR')
  00249          .  .  .  (3) CALL TEXT ('SURFACE CONTOURING')
  00250          .  .  .  (4) CALL TEXT ('MATRIX EDITOR')
  00251          .  .  .  (5) CALL TEXT ('LOCATE ZEROS')
  00252          .  .  .  (6) CALL TEXT ('FILL ZEROS')
  00253          .  .  .  (7) CALL TEXT ('RETURN TO CIRMIS MONITOR')
  00254          .  .  ...FIN
  00255          .  .  IY = IY-40
  00256          .  ...FIN
  00257          .  CALL APOS (50,210)
  00258          .  CALL BOX (925,310)
  00259          .  IX = 200
  00260          .  IY = 500
  00261          .  CALL APOS (IX,IY)
  00262          .  CALL TEXT ('GENERAL PROJECT SPECIFICATIONS (USER INPUT)*')
  00263          .  CALL APOS (50,IY-6)
  00264          .  CALL RVEC (925,0)
  00265          .  IX = 140
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:02:53  PAGE  00006
  00266          .  IY = IY-50
  00267          .  DO (J=21,26,1)
  00268          .  .  CALL NAME (J)
  00269          .  .  CALL APOS (IX,IY)
  00270          .  .  SELECT (J)
  00271          .  .  .  (21) CALL TEXT ('NODES IN X DIRECTION (I3)       =')
  00272          .  .  .  (22) CALL TEXT ('NODES IN Y DIRECTION (I3)       =')
  00273          .  .  .  (23) CALL TEXT ('X NODAL INCREMENT (F10.0)       =')
  00274          .  .  .  (24) CALL TEXT ('Y NODAL INCREMENT (F10.0)       =')
  00275          .  .  .  (25) CALL TEXT ('MAP NAME & EXTENSION (30A1)     =')
  00276          .  .  .  (26) CALL TEXT ('SURFACE NAME & EXTENSION (30A1) =')
  00277          .  .  ...FIN
  00278          .  .  IY = IY-40
  00279          .  ...FIN
  00280          .  CALL LPHIT (0)
  00281          .  CALL CHRSCL (0)
  00282          .  CALL APOS (510,170)
  00283          .  CALL TEXT ('KEYBOARD INPUT:')
  00284          .  CALL APOS (30,150)
  00285          .  CALL TEXT ('**FILE NAMES MUST USE 11/55 FILE NAMING',-15)
  00286          .  CALL TEXT ('  CONVENTION,  (I.E.  DEV:[UIC]FNAME.EXT)')
  00287          .  CALL ENDSB(ISZ)
  00288    C     .  WRITE(5,5777) ISZ
  00289    5777  .  FORMAT(' ISZ = ',I5)
  00290         ...FIN
  00291         END

-------------------------------------------

      PROCEDURE CROSS-REFERENCE TABLE

00180 DETERMINE-MAP-DISPLAY-FILE-NAME
      00107

00195 DISPLAY-FILE-SPECIFICATIONS
      00079  00162

00169 DISPLAY-STAR
      00089

00219 SET-UP-DISPLAY-FILES
      00037


(FLECS VERSION 22.46)

-------------------------------------------
```

MAPDRW.FLX

MAP GENERATOR

```
00001    C  *******   [351,101] MAPDRW.FLX   ******
00002    C
00003    C  DIGITIZER PROGRAM TO OUTPUT MAP BOUNDARYS.
00004    C  MAY BE USED WITH HASH AND WITH CALCOMP.
00005    C
00006    C
00007    C
00008    C
00009    C   PROGRAM OPTIONS:
00010    C
00011    C    TELETYPE KEYBOARD:
00012    C     PEN NO.---1,2 OR 3
00013    C     OPTION---- 1 BOUNDARY LINES
00014    C     OPTION---- 2 LABELS
00015    C     OPTION---- 3 SINGLE CHARACTER(*)
00016    C     OPTION---- 4 VECTORS
00017    C
00018    C    DIGITIZER KEYBOARD:
00019    C
00020    C
00021          BYTE FNAME,SRCMAP,DSPMAP
00022          BYTE CSTRNG(30),KEYBUF(30),EL,RR,YY
00023    C
00024          INTEGER DISPNT,BKGRND
00025          REAL MAXMAP
00026          LOGICAL NEWMAP
00027    C
00028          DIMENSION  NDIGIT(72), ISIZ(17)
00029          DIMENSION X(500),Y(500)
00030    C
00031          COMMON     MAP(3000), LINES(1000)
00032          COMMON /DSP1/ IMAIN(200),LABEL(600),LPOPT(80),DISPNT(50)
00033          COMMON /DSP2/ MESS1(60),MESS2(50),MESS21(50),MESS3(70),MESS4(40)
00034         1       ,MESS5(45),MESS6(45),MESS7(40),MESS8(30),MESS9(25)
00035         2             ,MESS10(50),MESS11(30),MESS22(40)
00036          COMMON /DSP3/ BKGRND(20), ISPECS(100)
00037    C
00038          COMMON/TND/ INITDZ,DZXMAX,DZYMAX,DZMAX,NODESX,NODESY,XINC,YINC,
00039         1            NSX,TOTSEC,TOTX,FNAME(30),SRCMAP(30),DSPMAP(30),
00040         2            XMAX,YMAX,MAXMAP,DSMAX
00041    C
00042          EQUIVALENCE (INITDZ,NDIGIT(1))
00043    C
00044    C
00045          DATA EL,RR,YY/'L','R','Y'/
00046    C
00047    C
00048    C
00049          NEWMAP = .FALSE.
00050          LMN = 0
00051          IMN = 0
00052          MSIZE = 0
00053          SIZ = 0.07
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00002
  00054          ROT = 0.0
  00055          ISTAR = 11
  00056    C
  00057    C*************************************************************
  00058    C
  00059    C  ********  INITIALIZE DISPLAY AND DIGITIZER  ********
  00060    C
  00061    C*************************************************************
  00062    C
  00063          CALL MAINOF (IMAIN,200,0,1)
  00064          CALL CLKOFF
  00065          CALL BXINIT (2,1)
  00066    1     CALL BXSWCH (ISW)
  00067          IF (ISW.NE.1)
  00068          .  WRITE (5,5001)
  00069    5001  .  FORMAT (//' SET INC/ABS SWITCH ON DIGITIZER CONSOLE TO ABS'//)
  00070          .  GO TO 1
  00071          ...FIN
  00072    C
  00073    C
  00074    C
  00075          SET-UP-SUBDISPLAY-FILES
  00076    C
  00077    C
  00078    C
  00079    C
  00080    C
  00081    C*************************************************************
  00082    C
  00083    C  **************  START PROGRAM  **************
  00084    C
  00085    C*************************************************************
  00086    C
  00087    C
  00088          CALL OPFILE (4,'SURFSPC',1,,IPNT0)
  00089          CALL OPR (IPNT0,0,,NOIGIT,72.)
  00090    C
  00091          CALL SUBOF (ISPEC8,100,0)
  00092          CALL AREA   (1)
  00093          CALL INTENS (7)
  00094          CALL CHRTYP (0,0,0)
  00095          CALL CHRSCL (0)
  00096          CALL APOS   (50,945)
  00097          CALL INNBR(NODESX,'(I4)',-15)
  00098          CALL INNBR(NODESY,'(I4)',-40)
  00099          WHEN (XINC.LT.10000.)
  00100          .  CALL FNMBR(XINC,'(F8.3)',-15)
  00101          .  CALL FNMBR(YINC,'(F8.3)',-40)
  00102          ...FIN
  00103          ELSE
  00104          .  CALL FNMBR(XINC,'(1PE12.5)',-15)
  00105          .  CALL FNMBR(YINC,'(1PE12.5)',-40)
  00106          ...FIN
  00107          NCHAR = ICHR(SRCMAP)
  00108          CALL TEXT (SRCMAP,1,NCHAR)
  00109          CALL AREA (0)
```

```
(FLECS VERSION 22.46)      14-SEP-79   08:47:31  PAGE  0003
  00110          CALL ENDSB (ISIZ(16))
  00111    C
  00112    C
  00113    C
  00114    C  ***  INITIALIZE DIGITIZER  ***
  00115    C
  00116          IF (INITDZ.EQ.0)
  00117         .  CALL STRTSB (MESS1)
  00118    1000 .  CALL BX1XY (DX0,DY0,IBTN)
  00119         .  IF (IBTN.NE.1) GO TO 1000
  00120         .  CALL BXBEEP
  00121         .  CALL STOPSB (MESS1)
  00122         .  CALL STRTSB (MESS2)
  00123    1001 .  CALL BX1XY (DZX,DZY,IBTN)
  00124         .  IF (IBTN.NE.2) GO TO 1001
  00125         .  CALL BXBEEP
  00126         .  CALL STOPSB (MESS2)
  00127    C  INPUT DISTANCE ON MAP AND CALCULATE MAXIMUM INCHES FOR DIGITIZER
  00128         .  CALL STRTSB (MESS21)
  00129         .  CALL BXKEY (KEYBUF,30,ICC,NCHAR)
  00130         .  DECODE (NCHAR,225,KEYBUF) SCALE
  00131    225  .  FORMAT(F10.0)
  00132         .  CALL STOPSB (MESS21)
  00133         .  DZXMAX = XMAX/(SCALE/DZX)
  00134         .  DZYMAX = YMAX/(SCALE/DZX)
  00135         .  INITDZ = 1
  00136         ...FIN
  00137          DZMAX = AMAX1(DZXMAX,DZYMAX)
  00138    C  DISPLAY MAP BOUNDARY
  00139          IXM=XMAX*DSMAX/MAXMAP
  00140          IYM=YMAX*DSMAX/MAXMAP
  00141          CALL SUBUF(BKGRND,20,0)
  00142          CALL INTENS(7)
  00143          CALL APOS(0,0)
  00144          CALL AVEC(IXM,0)
  00145          CALL AVEC(IXM,IYM)
  00146          CALL AVEC(0,IYM)
  00147          CALL AVEC(0,0)
  00148          CALL ENDSB (ISIZ(17))
  00149          CALL STRTSB (BKGRND)
  00150    C     WRITE (5,520) ISIZ
  00151    520   FORMAT (' SUBPICTURE SIZES = ',17I5)
  00152    C
  00153    C*************************************************************
  00154    C
  00155    C     ********  DISPLAY GRAPHICS INTERRUPTS  ********
  00156    C
  00157    C*************************************************************
  00158    C
  00159    20    CONTINUE
  00160          CALL GRATTN  (1,ITYP,'LP')
  00161          IF (ITYP.NE.'LP') GO TO 20
  00162    C
  00163    C  LIGHT PEN INTERRUPTS
  00164          CALL LTPEN (ID,ITIP)
  00165          IF (ITIP.EQ.1)
```

```
(FLECS VERSION 22,46)     14-SEP-79   08:47:31  PAGE  00004
  00166          ,  SELECT (ID)
  00167    C     .  .
  00168    C    RETURN CALLING PROGRAM
  00169    C     .  .
  00170          .  .  (1)
  00171          .  .  .  IF (NEWMAP)
  00172          .  .  .  .  CALL DPW (IPNT0,0,,NDIGIT,72,)
  00173          .  .  .  .  CLOSE(UNIT=6)
  00174          .  .  .  .  CALL SAVSOF (3,DSPMAP,MAP)
  00175          .  .  .  ...FIN
  00176          .  .  .  TASK = RAD50('APPLIC')
  00177          .  .  .  CALL REQUES (TASK,,IDS)
  00178          .  .  ...FIN
  00179    C     .  .
  00180    C   START DIGITIZING
  00181    C     .  .
  00182          .  .  (2)
  00183          .  .  .  WHEN (NEWMAP)
  00184          .  .  .  .  CALL STRTSB (MESS11)
  00185          .  .  .  .  CALL WAIT (10,2,MMM)
  00186          .  .  .  .  CALL STOPSB (MESS11)
  00187          .  .  .  ...FIN
  00188          .  .  .  ELSE
  00189          .  .  .  .  OPEN (UNIT=6,NAME=SRCMAP,TYPE='NEW')
  00190          .  .  .  .  NEWMAP=.TRUE.
  00191          .  .  .  .  CALL STRTSB (MESS22)
  00192          .  .  .  .  CALL WAIT (5,2,MM)
  00193          .  .  .  .  CALL STOPSB (MESS22)
  00194          .  .  .  .  IMN = 0
  00195          .  .  .  .  CALL STOPSB (BKGRND)
  00196          .  .  .  .  CALL STOPSB (LPOPT)
  00197          .  .  .  .  CALL SUBDF (MAP,3000,0)
  00198          .  .  .  .  CALL INTENS(7)
  00199          .  .  .  .  CALL APOS(0,0)
  00200          .  .  .  .  CALL AVEC(IXM,0)
  00201          .  .  .  .  CALL AVEC(IXM,IYM)
  00202          .  .  .  .  CALL AVEC(0,IYM)
  00203          .  .  .  .  CALL AVEC(0,0)
  00204          .  .  .  .  CALL AREA (0)
  00205          .  .  .  .  CALL ENDSB
  00206          .  .  .  .  IF (IO.EQ.2) GO TO 40
  00207          .  .  .  ...FIN
  00208          .  .  ...FIN
  00209          .  .  (3)
  00210          .  .  .  IF (NEWMAP) CLOSE (UNIT=6,DISPOSE='DELETE')
  00211          .  .  .  TASK = RAD50('CIRMIS')
  00212          .  .  .  CALL REQUES (TASK,,IDS)
  00213          .  .  ...FIN
  00214          .  ...FIN
  00215          .  IF ( IO.EQ.1. OR .IO.EQ.3) CALL EXIT
  00216          ...FIN
  00217          GO TO 20
  00218    C
  00219    C
  00220    C*************************************************************
  00221    C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08147131  PAGE  00005
  00222   C************* DIGITIZER INTERRUPTS ***************
  00223   C
  00224   C*************************************************************
  00225   C
  00226   40     CONTINUE
  00227          CALL STRTSB (MESS3)
  00228          CALL BXKEY (KEYBUF,30,ICC,NCHAR)
  00229          DECODE(NCHAR,740,KEYBUF) IPEN,IOPT
  00230   740    FORMAT(2I1)
  00231          CALL STOPSB (MESS3)
  00232          SELECT (IOPT)
  00233          .  (1) CALL STRTSB(MESS4)
  00234          .  (2)
  00235          .  .  CALL STRTSB (MESS5)
  00236          .  .  CALL BXKEY  (KEYBUF,30,ICC,NCHAR)
  00237          .  .  DECODE(NCHAR,741,KEYBUF) CSTRNG
  00238   741    .  .  FORMAT(30A1)
  00239          .  .  CALL STOPSB (MESS5)
  00240          .  .  CALL STRTSB (MESS6)
  00241          .  ...FIN
  00242          .  (3) CALL STRTSB (MESS7)
  00243          .  (4) CALL STRTSB (MESS8)
  00244          ...FIN
  00245          IF (IOPT.EQ.0 .OR. IPEN.EQ.0)
  00246          .  CALL STRTSB (LPOPT)
  00247          .  GO TO 20
  00248          ...FIN
  00249   C
  00250   C  *********  DIGITIZE POINTS  **********
  00251   C
  00252          II=0
  00253          CALL SUBDF(LINES,1000,IMN)
  00254          LMN = 0
  00255          IMN = 2
  00256          INT = IPEN*2 + 1
  00257          CALL INTENS(INT)
  00258          CALL LINTYP(0)
  00259          CALL CHRTYP(0,0,0)
  00260          CALL CHRSCL(0)
  00261          MORE = .TRUE.
  00262          REPEAT WHILE(MORE)
  00263          .  CALL BX1XY(DZX,DZY,IBUT)
  00264          .  CALL BXLON
  00265          .  CALL BXBEEP
  00266   C    SELECT ON SPECIFIC CURSOR BUTTONS
  00267          .  SELECT (IBUT)
  00268   C *** WHITE---DIGITIZE POINT
  00269          .  .  (2)
  00270          .  .  .  II=II+1
  00271          .  .  .  WHEN(II.GT.500)
  00272          .  .  .  .  CALL STRTSB(MESS9)
  00273          .  .  .  .  DO(L=1,10) CALL BXBEEP
  00274          .  .  .  .  CALL WAIT(10,2,MMM)
  00275          .  .  .  .  CALL STOPSB(MESS9)
  00276          .  .  .  .  II=II-1
  00277          .  .  .  .  WRITE-LINE-TO-DISC-FILE
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00006
  00278          .  .  .  .  MORE = .FALSE.
  00279          .  .  .  ...FIN
  00280          .  .  .  ELSE
  00281          .  .  .  .  CALCULATE-GRAPHIC-AND-REAL-VALUES
  00282          .  .  .  .  SELECT (IOPT)
  00283    C   LINES
  00284          .  .  .  .  .  (1)
  00285          .  .  .  .  .  .  DISPLAY-NUMBER-OF-POINTS-USED
  00286          .  .  .  .  .  .  WHEN (II.EQ.1)
  00287          .  .  .  .  .  .  .  CALL APOS(IX,IY)
  00288          .  .  .  .  .  .  .  CALL STOPSB (MESS4)
  00289          .  .  .  .  .  .  ...FIN
  00290          .  .  .  .  .  .  ELSE
  00291          .  .  .  .  .  .  .  CALL AVEC(IX,IY)
  00292          .  .  .  .  .  .  ...FIN
  00293          .  .  .  .  .  ...FIN
  00294    C   CHARACTER STRING
  00295          .  .  .  .  .  (2)
  00296          .  .  .  .  .  .  WRITE-LINE-TO-DISC-FILE
  00297          .  .  .  .  .  .  CALL STOPSB (MESS6)
  00298          .  .  .  .  .  .  MORE=.FALSE.
  00299          .  .  .  .  .  ...FIN
  00300    C   SINGLE CHARACTER
  00301          .  .  .  .  .  (3)
  00302          .  .  .  .  .  .  WRITE-LINE-TO-DISC-FILE
  00303          .  .  .  .  .  .  CALL STOPSB (MESS7)
  00304          .  .  .  .  .  .  MORE = .FALSE.
  00305          .  .  .  .  .  ...FIN
  00306    C   VECTORS
  00307          .  .  .  .  .  (4)
  00308          .  .  .  .  .  .  WHEN(MOD(II,2).NE.0) CALL APOS(IX,IY)
  00309          .  .  .  .  .  .  ELSE
  00310          .  .  .  .  .  .  .  CALL AVEC(IX,IY)
  00311          .  .  .  .  .  .  ...FIN
  00312          .  .  .  .  .  .  CALL STOPSB (MESS8)
  00313          .  .  .  .  .  ...FIN
  00314          .  .  .  .  ...FIN
  00315          .  .  .  ...FIN
  00316          .  .  .  CALL BXLOFF
  00317          .  .  ...FIN
  00318    C *** BLUE----STORE LINE OR VECTOR DATA IN DISC FILE
  00319          .  .  (3)
  00320          .  .  .  WRITE-LINE-TO-DISC-FILE
  00321          .  .  .  MORE = .FALSE.
  00322          .  .  ...FIN
  00323    C *** GREEN---DELETE LAST POINT
  00324          .  .  (4)
  00325          .  .  .  CALL ENDSB (LINES)
  00326          .  .  .  CALL SUBDF(LINES,1000,2)
  00327          .  .  .  CALL INTENS(INT)
  00328          .  .  .  II = II-1
  00329          .  .  .  WHEN (II.LE.0) MORE = .FALSE.
  00330          .  .  .  ELSE
  00331          .  .  .  .  DO (L=1,II)
  00332          .  .  .  .  .  IX = X(L)*DSMAX/MAXMAP
  00333          .  .  .  .  .  IY = Y(L)*DSMAX/MAXMAP
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00007
  00334          .  .  .  .  .  SELECT (IOPT)
  00335          .  .  .  .  .  .  (1)
  00336          .  .  .  .  .  .  .  IF(L.EQ.1) CALL APOS (IX,IY)
  00337          .  .  .  .  .  .  .  IF(L.NE.1) CALL AVEC (IX,IY)
  00338          .  .  .  .  .  .  ...FIN
  00339          .  .  .  .  .  .  (4)
  00340          .  .  .  .  .  .  .  IF(MOD(L,2).NE.0) CALL APNT(IX,IY)
  00341          .  .  .  .  .  .  .  IF(MOD(L,2).EQ.0) CALL AVEC(IX,IY)
  00342          .  .  .  .  .  .  ...FIN
  00343          .  .  .  .  .  ...FIN
  00344          .  .  .  .  ...FIN
  00345          .  .  .  ...FIN
  00346          .  .  ...FIN
  00347          .  ...FIN
  00348          ...FIN
  00349          GO TO 40

         ----------------------------------------

  00350          TO CALCULATE-GRAPHIC-AND-REAL-VALUES
  00351    C     .  IF(DZX.LT.0) DZX=0.
  00352    C     .  IF(DZX.GT.DZXMAX) DZX=DZXMAX
  00353    C     .  IF(DZY.LT.0) DZY=0
  00354    C     .  IF(DZY.GT.DZYMAX) DZY=DZYMAX
  00355    C     .  IF(IABS(DZX-DZXL).LT.1) DZX=DZXL
  00356    C     .  IF(IABS(DZY-DZYL).LT.1) DZY=DZYL
  00357          .  X(II) = DZX*MAXMAP/DZMAX
  00358          .  Y(II) = DZY*MAXMAP/DZMAX
  00359          .  IX    = X(II)*DSMAX/MAXMAP
  00360          .  IY    = Y(II)*DSMAX/MAXMAP
  00361          ...FIN

         ----------------------------------------

  00362          TO WRITE-LINE-TO-DISC-FILE
  00363          .  CALL ENDSB
  00364          .  CALL STOPSB (LINES)
  00365          .  CALL OPENSB(MAP,2000)
  00366          .  CALL INTENS(INT)
  00367          .  CALL LINTYP (0)
  00368          .  CALL CHRTYP (0,0,0)
  00369          .  CALL CHRSCL (0)
  00370          .  WHEN(IOPT.EQ.1.OR.IOPT.EQ.4)
  00371    C*** LINE OR VECTOR MODE ***
  00372          .  .  WRITE(6,601) II,IPEN,IOPT
  00373    601   .  .  FORMAT(3I5)
  00374          .  .  IF(MAXMAP.GE.10000.) WRITE(6,602) (X(L),Y(L),L=1,II)
  00375          .  .  IF(MAXMAP.LT.10000.) WRITE(6,603) (X(L),Y(L),L=1,II)
  00376    602   .  .  FORMAT(8F10.1)
  00377    603   .  .  FORMAT(8F10.4)
  00378          .  .  DO (L=1,II)
  00379          .  .  .  IX = X(L)*DSMAX/MAXMAP
  00380          .  .  .  IY = Y(L)*DSMAX/MAXMAP
  00381          .  .  .  SELECT (IOPT)
  00382          .  .  .  .  (1)
  00383          .  .  .  .  .  IF(L.EQ.1) CALL APOS (IX,IY)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00008
  00384          .  .  .  .  .  IF(L.NE.1) CALL AVEC (IX,IY)
  00385          .  .  .  .  ...FIN
  00386          .  .  .  .  (4)
  00387          .  .  .  .  .  IF(MOD(L,2).NE.0) CALL APNT(IX,IY)
  00388          .  .  .  .  .  IF(MOD(L,2).EQ.0) CALL AVEC(IX,IY)
  00389          .  .  .  .  ...FIN
  00390          .  .  .  ...FIN
  00391          .  .  ...FIN
  00392          .  ...FIN
  00393    C     .
  00394    C     .
  00395          .  ELSE
  00396          .  .  SELECT (IOPT)
  00397          .  .  .  (2)
  00398    C  CHARACTER STRING
  00399          .  .  .  .  WRITE(6,653) EL,IPEN,(CSTRNG(I),I=1,NCHAR)
  00400    653   .  .  .  .  FORMAT(A1,I4,30A1)
  00401          .  .  .  .  IF(MAXMAP.GE.10000.) WRITE(6,654) YY,X(1),Y(1),SIZ,ROT
  00402          .  .  .  .  IF(MAXMAP.LT.10000.) WRITE(6,655) YY,X(1),Y(1),SIZ,ROT
  00403    654   .  .  .  .  FORMAT(A1,4X,4F10.1)
  00404    655   .  .  .  .  FORMAT(A1,4X,4F10.4)
  00405          .  .  .  .  IX = X(1)*DSMAX/MAXMAP
  00406          .  .  .  .  IY = Y(1)*DSMAX/MAXMAP
  00407          .  .  .  .  CALL APOS(IX,IY)
  00408          .  .  .  .  CALL TEXT(CSTRNG,0,NCHAR)
  00409          .  .  .  ...FIN
  00410    C  SINGLE CHARACTER
  00411          .  .  .  (3)
  00412          .  .  .  .  WRITE(6,656) RR,IPEN,ISTAR
  00413    656   .  .  .  .  FORMAT(A1,I4,I5)
  00414          .  .  .  .  IF(MAXMAP.GE.10000.) WRITE(6,654) YY,X(1),Y(1),SIZ,ROT
  00415          .  .  .  .  IF(MAXMAP.LT.10000.) WRITE(6,655) YY,X(1),Y(1),SIZ,ROT
  00416          .  .  .  .  IX = X(1)*DSMAX/MAXMAP - 2.
  00417          .  .  .  .  IY = Y(1)*DSMAX/MAXMAP - 5.
  00418          .  .  .  .  CALL APOS(IX,IY)
  00419          .  .  .  .  CALL TEXT('*',0)
  00420          .  .  .  ...FIN
  00421          .  .  ...FIN
  00422          .  ...FIN
  00423          .  CALL ENDSR(MSIZE)
  00424          .  LMN = 0
  00425          .  DISPLAY-NUMBER-OF-POINTS-USED
  00426          .  IF (MSIZE.GT.2800) DISPLAY-OVERFLOW-WARNING
  00427          .  CALL STRTSR (MAP)
  00428          ...FIN

         ----------------------------------------

  00429          TO DISPLAY-NUMBER-OF-POINTS-USED
  00430          .  CALL SUBDF (DISPNT,50,LMN)
  00431          .  LMN = 2
  00432          .  CALL AREA (1)
  00433          .  CALL INTENS (4)
  00434          .  CALL CHRTYP (0,0,0)
  00435          .  CALL CHRSCL (0)
  00436          .  CALL APOS (131,85)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00009
  00437          .  CALL INMBR (II,'(I4)')
  00438          .  CALL APOS (131,20)
  00439          .  CALL INMBR (MSIZE,'(I4)')
  00440          .  CALL AREA (0)
  00441          .  CALL ENDSR
  00442    C     .  CALL STRTSR (DISPNT)
  00443          ...FIN

         ---------------------------------------

  00444          TO DISPLAY-OVERFLOW-WARNING
  00445          .  CALL STRTSR (MESS10)
  00446          .  DO (J=1,10)
  00447          .  .  CALL BXBEEP
  00448          .  .  CALL WAIT(1,2,MMM)
  00449          .  ...FIN
  00450          .  CALL WAIT(2,2,MMM)
  00451          .  CALL STOPSR(MESS10)
  00452          ...FIN

         ---------------------------------------

  00453          TO SET-UP-SUBDISPLAY-FILES
  00454    C     .
  00455    C*************************************************************
  00456    C     .
  00457    C     .
  00458    C **********  MAJOR PROGRAM SUBDISPLAYS **********
  00459    C     .
  00460    C     .
  00461    C*************************************************************
  00462    C     .
  00463    C     .
  00464          .  CALL SUBOF (LABEL,600,1)
  00465          .  CALL INTENS (7)
  00466          .  CALL CHRTYP (1,0,0)
  00467          .  CALL CHRSCL (3)
  00468          .  CALL APOS (60,1000)
  00469          .  CALL TEXT('MAP BOUNDARY DIGITIZING ROUTINE')
  00470          .  CALL AREA(1)
  00471          .  CALL CHRSCL (0)
  00472          .  CALL APOS (0,120)
  00473          .  CALL TEXT ('LINE ARRAY',-2)
  00474          .  CALL RVEC (70,0)
  00475          .  CALL APOS (0,100)
  00476          .  CALL TEXT ('POINTS AVAILABLE =  500',-15)
  00477          .  CALL TEXT ('POINTS USED      = ',-30)
  00478          .  CALL TEXT ('MASTER DISPLAY FILE',-2)
  00479          .  CALL RVEC (130,0)
  00480          .  CALL APOS (0,35)
  00481          .  CALL TEXT ('POINTS AVAILABLE = 3000',-15)
  00482          .  CALL TEXT ('POINTS USED      = ')
  00483          .  CALL CHRSCL (1)
  00484          .  CALL APOS (10,1000)
  00485          .  CALL TEXT('SPECIFICATIONS')
  00486          .  CALL APOS (0,993)
```

```
(FLECS VERSION 22.46)      14-SEP-79   08:47:31  PAGE  00010
  00487          .  CALL RVEC (205,0)
  00488          .  CALL RVEC (0,30)
  00489          .  CALL RVEC (-205,0)
  00490          .  CALL RVEC (0,-30)
  00491          .  CALL RPOS (2,2)
  00492          .  CALL RVEC (201,0)
  00493          .  CALL RVEC (0,26)
  00494          .  CALL RVEC (-201,0)
  00495          .  CALL RVEC (0,-26)
  00496          .  CALL CHRSCL (0)
  00497          .  CALL APOS (0,960)
  00498          .  CALL TEXT('NUMBER OF NODES:',-15)
  00499          .  CALL TEXT('     X=',-15)
  00500          .  CALL TEXT('     Y=',-25)
  00501          .  CALL TEXT('NODAL INCREMENTS:',-15)
  00502          .  CALL TEXT('     X=',-15)
  00503          .  CALL TEXT('     Y=',-25)
  00504          .  CALL TEXT('MAP FILE NAME:')
  00505          .  CALL CHRSCL (1)
  00506          .  CALL APOS (70,800)
  00507          .  CALL TEXT ('CONTROLS')
  00508          .  IX = 0
  00509          .  IY = 793
  00510          .  CALL APOS (IX,IY)
  00511          .  CALL RVEC (220,0)
  00512          .  CALL RVEC (0,30)
  00513          .  CALL RVEC (-220,0)
  00514          .  CALL RVEC (0,-30)
  00515          .  CALL RPOS (2,2)
  00516          .  CALL RVEC (216,0)
  00517          .  CALL RVEC (0,26)
  00518          .  CALL RVEC (-216,0)
  00519          .  CALL RVEC (0,-26)
  00520          .  CALL INTENS (5)
  00521          .  CALL CHRSCL (0)
  00522          .  CALL APOS (30,750)
  00523          .  CALL TEXT ('VS60 DISPLAY',-3)
  00524          .  CALL RVEC (82,0)
  00525          .  CALL APOS (14,710)
  00526          .  CALL TEXT ('LIGHT PEN OPTIONS:')
  00527          .  CALL APOS (45,530)
  00528          .  CALL TEXT ('DIGITIZER',-3)
  00529          .  CALL RVEC (60,0)
  00530          .  CALL APOS (0,500)
  00531          .  CALL TEXT ('CURSOR BUTTONS:',-17)
  00532          .  CALL TEXT ('       RED --- INITIALIZE REGION',-17)
  00533          .  CALL TEXT ('       WHITE - DIGITIZE POINT',-17)
  00534          .  CALL TEXT ('       BLUE -- WRITE LINE TO DISC FILE',-17)
  00535          .  CALL TEXT ('       GREEN - DELETE LAST POINT',-17)
  00536          .  CALL TEXT ('       YELLOW- HOLD POSITION ON MAP',-30)
  00537          .  CALL TEXT ('KEYBOARD OPTIONS:',-17)
  00538          .  CALL TEXT ('       PEN NO. --- 1,2 OR 3',-17)
  00539          .  CALL TEXT ('       OPTIONS:',-20)
  00540          .  CALL TEXT ('                1 - BOUNDARY LINES',-17)
  00541          .  CALL TEXT ('                2 - LABELS',-17)
  00542          .  CALL TEXT ('                3- SINGLE CHARACTER(*)',-17)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00011
  00543           .  CALL TEXT ('               4 = VECTORS',-15)
  00544    C      .
  00545           .  CALL TEXT ('OTHER KEYBOARD INPUT',-15)
  00546           .  CALL TEXT ('AS REQUESTED!')
  00547    C      .
  00548           .  CALL AREA (0)
  00549           .  CALL ENDSB(TSIZ(1))
  00550           .  CALL STRTSB (LABEL)
  00551    C      .
  00552           .  CALL SUBDF (LPOPT,80,0)
  00553           .  CALL AREA (1)
  00554           .  CALL INTENS (5)
  00555           .  CALL CHRTYP (0,0,0)
  00556           .  CALL CHRSCL (0)
  00557           .  CALL MENU(680,40,1,'    RETURN TO CONTROL',
  00558          1.  '   START DIGITIZING','       ABORT & RETURN TO CONTROL')
  00559           .  CALL AREA (0)
  00560           .  CALL ENDSB(TSIZ(2))
  00561    C      .
  00562    C*****************************************************************
  00563    C      .
  00564    C  ******* DIGITIZER SUBDISPLAY FILE  *******
  00565    C      .
  00566    C*****************************************************************
  00567    C      .
  00568           .  CALL SUBDF (MESS1,60,1)
  00569           .  CALL INTENS (7)
  00570           .  CALL CHRSCL (1)
  00571           .  CALL BLINK (1)
  00572           .  CALL APOS (75,500)
  00573           .  CALL TEXT ('*')
  00574           .  CALL BLINK (0)
  00575           .  CALL APOS (100,500)
  00576           .  CALL TEXT('DEPRESS RED BUTTON AT (0,0) ON MAP',-30)
  00577           .  CALL TEXT('(THIS IS ALSO START OF MAP SCALE)')
  00578           .  CALL ENDSB (ISIZ(3))
  00579    C      .
  00580           .  CALL SUBDF (MESS2,50,1)
  00581           .  CALL INTENS (7)
  00582           .  CALL CHRSCL (1)
  00583           .  CALL BLINK (1)
  00584           .  CALL APOS (5,500)
  00585           .  CALL TEXT ('*')
  00586           .  CALL BLINK (0)
  00587           .  CALL APOS (30,500)
  00588           .  CALL TEXT('DEPRESS WHITE BUTTON AT END OF SPECIFIED',0)
  00589           .  CALL TEXT(' HORIZONTAL SCALE ON MAP')
  00590           .  CALL ENDSB (ISIZ(4))
  00591    C      .
  00592           .  CALL SUBDF (MESS21,50,1)
  00593           .  CALL INTENS (7)
  00594           .  CALL CHRSCL (1)
  00595           .  CALL BLINK (1)
  00596           .  CALL APOS (75,500)
  00597           .  CALL TEXT ('*')
  00598           .  CALL BLINK (0)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  0012
  00599           .  CALL APOS (100,500)
  00600           .  CALL TEXT('ENTER MAP SCALE DISTANCE ON DIGITIZER KEYBOARD',0)
  00601           .  CALL TEXT('  (F10.0)')
  00602           .  CALL ENDSR (ISIZ(5))
  00603   C       .
  00604           .  CALL SUBDF (MESS22,40,1)
  00605           .  CALL INTENS (3)
  00606           .  CALL CHRTYP (0,0,0)
  00607           .  CALL CHRSCL (1)
  00608           .  CALL BLINK (1)
  00609           .  CALL APOS (10,970)
  00610           .  CALL TEXT ('NEW MAP FILES HAVE BEEN INITIALIZED ON DISC')
  00611           .  CALL ENDSR (ISIZ(6))
  00612   C       .
  00613           .  CALL SUBDF (MESS3,70,1)
  00614           .  CALL INTENS (3)
  00615           .  CALL CHRTYP (0,0,0)
  00616           .  CALL CHRSCL (1)
  00617           .  CALL BLINK (1)
  00618           .  CALL APOS (175,970)
  00619           .  CALL TEXT ('*')
  00620           .  CALL BLINK (0)
  00621           .  CALL APOS (200,970)
  00622           .  CALL TEXT('ENTER PEN NO. & OPTION ON DIGITIZER KEYBOARD',0)
  00623           .  CALL TEXT('  (2I1)',-30)
  00624           .  CALL TEXT('OR "0" TO TERMINATE SEQUENCE')
  00625           .  CALL ENDSR (ISIZ(7))
  00626   C       .
  00627           .  CALL SUBDF (MESS4,40,1)
  00628           .  CALL INTENS (3)
  00629           .  CALL CHRTYP (0,0,0)
  00630           .  CALL CHRSCL (1)
  00631           .  CALL BLINK (1)
  00632           .  CALL APOS (175,970)
  00633           .  CALL TEXT ('*')
  00634           .  CALL BLINK (0)
  00635           .  CALL APOS (200,970)
  00636           .  CALL APOS (200,970)
  00637           .  CALL TEXT ('DIGITIZE LINE (WHITE BUTTON)')
  00638           .  CALL ENDSR (ISIZ(8))
  00639   C       .
  00640           .  CALL SUBDF (MESS5,45,1)
  00641           .  CALL INTENS (3)
  00642           .  CALL CHRTYP (0,0,0)
  00643           .  CALL CHRSCL (1)
  00644           .  CALL BLINK (1)
  00645           .  CALL APOS (175,970)
  00646           .  CALL TEXT ('*')
  00647           .  CALL BLINK (0)
  00648           .  CALL APOS (200,970)
  00649           .  CALL TEXT ('ENTER CHARACTER STRING ON DIGITIZER KEYBOARD')
  00650           .  CALL ENDSR (ISIZ(9))
  00651   C       .
  00652   C       .
  00653   C       .
  00654           .  CALL SUBDF (MESS6,45,1)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  0013
  00655           .  CALL INTENS (3)
  00656           .  CALL CHRTYP (0,0,0)
  00657           .  CALL CHRSCL (1)
  00658           .  CALL BLINK (1)
  00659           .  CALL APOS (175,970)
  00660           .  CALL TEXT ('*')
  00661           .  CALL BLINK (0)
  00662           .  CALL APOS (200,970)
  00663           .  CALL TEXT ('DIGITIZE START LOCATION OF STRING (WHITE BUTTON)')
  00664           .  CALL ENDSR (ISIZ(10))
  00665   C       .
  00666           .  CALL SUBDF (MES$7,40,1)
  00667           .  CALL INTENS (3)
  00668           .  CALL CHRTYP (0,0,0)
  00669           .  CALL CHRSCL (1)
  00670           .  CALL BLINK (1)
  00671           .  CALL APOS (175,970)
  00672           .  CALL TEXT ('*')
  00673           .  CALL BLINK (0)
  00674           .  CALL APOS (200,970)
  00675           .  CALL TEXT ('DIGITIZE CENTER OF SYMBOL (WHITE BUTTON)')
  00676           .  CALL ENDSR (ISIZ(11))
  00677   C       .
  00678           .  CALL SUBDF (MES$8,30,1)
  00679           .  CALL INTENS (3)
  00680           .  CALL CHRTYP (0,0,0)
  00681           .  CALL CHRSCL (1)
  00682           .  CALL BLINK (1)
  00683           .  CALL APOS (175,970)
  00684           .  CALL TEXT ('*')
  00685           .  CALL BLINK (0)
  00686           .  CALL APOS (200,970)
  00687           .  CALL TEXT ('DIGITIZE VECTORS')
  00688           .  CALL ENDSR (ISIZ(12))
  00689   C       .
  00690           .  CALL SUBDF (MES$9,25,1)
  00691           .  CALL INTENS (3)
  00692           .  CALL CHRTYP (0,0,0)
  00693           .  CALL CHRSCL (1)
  00694           .  CALL BLINK (1)
  00695           .  CALL TEXT ('POINTS EXCEEDED---LINE SAVED')
  00696           .  CALL ENDSR (ISIZ(13))
  00697   C       .
  00698           .  CALL SUBDF (MES$10,50,1)
  00699           .  CALL INTENS (7)
  00700           .  CALL CHRSCL (2)
  00701           .  CALL BLINK (1)
  00702           .  CALL APOS (150,910)
  00703           .  CALL TEXT ('MASTER DISPLAY FIDLE IS ALMOST FULL')
  00704           .  CALL ENDSR (ISIZ(14))
  00705   C       .
  00706           .  CALL SUBDF (MES$11,30,1)
  00707           .  CALL INTENS (3)
  00708           .  CALL CHRSCL (1)
  00709           .  CALL BLINK (1)
  00710           .  CALL APOS (10,970)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:47:31  PAGE  00014
  00711           .  CALL TEXT ('SEQUENCE HAS ALREADY BEEN TERMINATED')
  00712           .  CALL ENDSR (ISIZ(15))
  00713    C      .
  00714           ...FIN
  00715           END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00350 CALCULATE-GRAPHIC-AND-REAL-VALUES
                00281

          00429 DISPLAY-NUMBER-OF-POINTS-USED
                00285  00425

          00444 DISPLAY-OVERFLOW-WARNING
                00426

          00453 SET-UP-SUBDISPLAY-FILES
                00275

          00362 WRITE-LINE-TO-DISC-FILE
                00277  00296  00302  00320


          (FLECS VERSION 22.46)

          ----------------------------------------
```

SURFGN.FLX

SURFACE GENERATOR

```
C       ********** [351,101]SURFGN.FLX  **********
C
C
C     PROGRAM TO DIGITIZE SURFACE CONTOURS, INTERPOLATE GRID
C     VALUES AND WRITE A BINARY MATRIX FILE.
C
C
C
C
C
C
      BYTE FNAME,SRCMAP,DSPMAP
      BYTE KEYBUF(30), OCODE ,ANS
C
      INTEGER CROSS
C
      REAL L1SQ, L2SQ,MAXMAP
C
      LOGICAL CLOSED, NORMAL, CIRCLE, RIVER, DONE
C
      DIMENSION X(400),Y(400),VALUE(10),NCOUNT(10),DIST(10)
      DIMENSION  NDIGIT(72), ISIZ(13)
C
      COMMON MAP(3000)
      COMMON /DTA/ SVAL(256)
      COMMON /DSP1/ MAIN(100),LABEL(520),LINES(900)
      COMMON /DSP2/ MESS1(55),MESS2(50),MESS21(45),MESS3(45),MESS4(45)
     1     ,MESS5(35),MESS6(25),MESS7(60),CROSS(15),MESS41(30)
     2     ,NPNT(30)
C
      COMMON/TNO/ INITDZ,DZXMAX,DZYMAX,DZMAX,NODESX,NODESY,XINC,YINC,
     1            NSX,TOTSEC,TOTX,FNAME(30),SRCMAP(30),DSPMAP(30),
     2            XMAX,YMAX,MAXMAP,DSMAX
C
      EQUIVALENCE (INITDZ,NDIGIT(1))
C
C*****************************************************************
C
C  ********  INITIALIZE DISPLAY AND DIGITIZER  ********
C
C*****************************************************************
C
      CALL MAINDF (MAIN,100,0,1,0,0,2)
      CALL CLKOFF
      CALL BXINIT (2,1)
1     CALL BXSWCH (ISW)
      IF (ISW.NE.1)
     .  WRITE (5,5001)
5001  .  FORMAT (///' SET INC/ABS SWITCH ON DIGITIZER CONSOLE TO ABS'//)
     .  GO TO 1
     ...FIN
C
C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00002
  00054    C
  00055           SET-UP-SUBDISPLAY-FILES
  00056    C
  00057    C
  00058    C
  00059    C
  00060    C
  00061    C
  00062    C*****************************************************************
  00063    C
  00064    C  ***************  START PROGRAM  ***************
  00065    C
  00066    C*****************************************************************
  00067    C
  00068    C
  00069           CALL OPFILE (4,'SURFSPC',1,,IPNT0)
  00070           CALL OPR (IPNT0,0,,NDIGIT,72,)
  00071           CALL OPFILE(4,'SURFTMP',TOTSEC,IPNT1)
  00072    D      WRITE(5,5111) NODESX,NODESY,XINC,YINC,NSX,TOTSEC,TOTX
  00073    5111   FORMAT(1X,2I5,2F10.1,I5,2F10.1)
  00074    C
  00075    C
  00076           SECOLD = -1000.
  00077           LMN=0
  00078    C
  00079    C  ***  INITIALIZE DIGITIZER  ***
  00080    C
  00081           IF (INITDZ.EQ.0)
  00082           .  CALL STRTSB (MESS1)
  00083    1000   .  CALL BX1XY (XX,YY,IBTN)
  00084           .  IF (IBTN.NE.1) GO TO 1000
  00085           .  CALL BXBEEP
  00086           .  CALL STOPSB (MESS1)
  00087           .  CALL STRTSB (MESS2)
  00088    1001   .  CALL BX1XY (DZX,DZY,IBTN)
  00089           .  IF (IBTN.NE.2) GO TO 1001
  00090           .  CALL BXBEEP
  00091           .  CALL STOPSB (MESS2)
  00092    C  INPUT DISTANCE ON MAP AND CALCULATE MAXIMUM INCHES FOR DIGITIZER
  00093           .  CALL STRTSB (MESS21)
  00094           .  CALL BXKEY (KEYBUF,30,ICC,NCHAR)
  00095           .  DECODE (NCHAR,225,KEYBUF) SCALE
  00096           .  CALL STOPSB (MESS21)
  00097           .  DZXMAX = XMAX/(SCALE/DZX)
  00098           .  DZYMAX = YMAX/(SCALE/DZX)
  00099           .  INITDZ = 1
  00100           ...FIN
  00101           DZMAX = AMAX1(DZXMAX,DZYMAX)
  00102    C  DISPLAY MAP BOUNDARY
  00103           IXM=XMAX*DSMAX/MAXMAP
  00104    D      WRITE (5,5114) SCALE,DZMAX,DZX,DZY
  00105    5114   FORMAT (' SCALE=',4F12.2)
  00106           IYM=YMAX*DSMAX/MAXMAP
  00107           WHEN (SRCMAP(1),NE.'N')
  00108           .  CALL RESSOF (3,DSPMAP,MAP)
  00109           .  CALL CALLSB (MAP)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00003
  00110          ...FIN
  00111          ELSE
  00112          .  CALL SUBDF(MAP,3000,0)
  00113          .  CALL INTENS(7)
  00114          .  CALL APOS(0,0)
  00115          .  CALL AVEC(IXM,0)
  00116          .  CALL AVEC(IXM,IYM)
  00117          .  CALL AVEC(0,IYM)
  00118          .  CALL AVEC(0,0)
  00119          .  CALL ENDSR
  00120          ...FIN
  00121    C  INITIALIZE TEMPORARY LINE DISPLAY FILE
  00122          CALL SUBDF(LINES,900,0)
  00123    C
  00124    C
  00125    C*****************************************************************
  00126    C
  00127    C  ***************  START DIGITIZING SEQUENCE  ***************
  00128    C
  00129    C*****************************************************************
  00130    C
  00131          DONE = .FALSE.
  00132    9     CONTINUE
  00133          START-NEW-SEQUENCE
  00134    D     WRITE (5,5114) SCALE,DZMAX,DZX,DZY
  00135    C
  00136    10    CONTINUE
  00137          REPEAT UNTIL (CLOSED)
  00138          .  CALL GRATTN (1,ITYP,'LP','DT')
  00139    C     .
  00140          .  SELECT (ITYP)
  00141          .  .  ('LP')
  00142    C  LIGHT PEN INTERRUPTS
  00143          .  .  .  CALL LTPEN (ID,ITIP)
  00144          .  .  .  IF(ITIP.EQ.1)
  00145          .  .  .  .  SELECT (ID)
  00146          .  .  .  .  .  (1)
  00147          .  .  .  .  .  .  CALL DPW (IPNT0,0,,NDIGIT,72.)
  00148          .  .  .  .  .  .  DONE = .TRUE.
  00149          .  .  .  .  .  ...FIN
  00150          .  .  .  .  .  (2)
  00151          .  .  .  .  .  .  ICH = ICHR(FNAME)
  00152          .  .  .  .  .  .  OPEN (UNIT=6,NAME=FNAME,TYPE='NEW',FORM='UNFORMATTED',ERR=99)
  00153          .  .  .  .  .  .  DO (J=1,NODESY)
  00154          .  .  .  .  .  .  .  ADR = (J-1)*NSX
  00155          .  .  .  .  .  .  .  CALL DPR (IPNT1,ADR,SVAL,TOTX)
  00156          .  .  .  .  .  .  .  WRITE (6) (SVAL(I),I=1,NODESX)
  00157          .  .  .  .  .  .  ...FIN
  00158          .  .  .  .  .  .  CLOSE (UNIT=6)
  00159          .  .  .  .  .  .  CALL DPW (IPNT0,0.,,NDIGIT,72.)
  00160          .  .  .  .  .  .  DONE = .TRUE.
  00161          .  .  .  .  .  ...FIN
  00162    C  ***  ABORT SEQUENCE  ***
  00163          .  .  .  .  .  (6)
  00164          .  .  .  .  .  .  CALL SUBDF (LINES,900,2)
  00165          .  .  .  .  .  .  START-NEW-SEQUENCE
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00004
  00166          .  .  .  .  .  ...FIN
  00167          .  .  .  .  .  (7)
  00168          .  .  .  .  .  .  CALL CLRDEV ('DT')
  00169          .  .  .  .  .  .  IF(ID.EQ.7) GO TO 9
  00170          .  .  .  .  .  ...FIN
  00171          .  .  .  .  ...FIN
  00172          .  .  .  .  IF( DONE ) GO TO 999
  00173          .  .  .  ...FIN
  00174          .  .  ...FIN
  00175          .  .  ('DT')
  00176    C DIGITIZER INTERRUPTS
  00177          .  .  .  CALL CURSOR (DZX,DZY,IBUT)
  00178          .  .  .  CALL BXLON
  00179          .  .  .  CALL BXBEEP
  00180    C     .  .  .
  00181          .  .  .  SELECT (IBUT)
  00182    C     .  .  .  .
  00183    C  *** WHITE--DIGITIZE POINT ***
  00184    C     .  .  .  .
  00185          .  .  .  .  (2)
  00186          .  .  .  .  .  JJ = JJ + 1
  00187          .  .  .  .  .  NPOINT = NPOINT + 1
  00188          .  .  .  .  .  DISPLAY-NUMBER-OF-POINTS-USED
  00189          .  .  .  .  .  STORE-POINTS-AND-CALCULATE-GRAPHIC-VALUES
  00190          .  .  .  .  .  NCOUNT(NLINE) = JJ
  00191    C     .  .  .  .  .  .
  00192          .  .  .  .  .  WHEN (JJ.EQ.1)
  00193    C     .  .  .  .  .  .  .
  00194          .  .  .  .  .  .  .  CALL STOPSB (MESS5)
  00195          .  .  .  .  .  .  .  DISPLAY-LINE-VALUE
  00196          .  .  .  .  .  .  .  FIRSTX = DZX
  00197          .  .  .  .  .  .  .  FIRSTY = DZY
  00198          .  .  .  .  .  .  ...FIN
  00199    C     .  .  .  .  .  .  .
  00200          .  .  .  .  .  .  ELSE
  00201          .  .  .  .  .  .  .  CALL AVEC (IDSX,IDSY)
  00202          .  .  .  .  .  .  ...FIN
  00203          .  .  .  .  .  ...FIN
  00204    C     .  .  .  .  .
  00205    C  *** BLUE -- TERMINATE LINE ***
  00206    C     .  .  .  .  .
  00207          .  .  .  .  (3)
  00208          .  .  .  .  .  CALL STRTSB (MESS4,MESS41)
  00209          .  .  .  .  .  CALL BXKEY (KEYBUF,30,ICC,NCHAR)
  00210          .  .  .  .  .  CALL STOPSB (MESS4,MESS41)
  00211          .  .  .  .  .  IF( NCHAR .LE. 4 )
  00212          .  .  .  .  .  .  DECODE (1,224,KEYBUF) ANS
  00213    224   .  .  .  .  .  .  FORMAT(A1)
  00214          .  .  .  .  .  .  IF (ANS.EQ.'D')
  00215          .  .  .  .  .  .  .  CLOSED = .TRUE.
  00216          .  .  .  .  .  .  .  CALL ENDSB
  00217          .  .  .  .  .  .  .  IF (CIRCLE)
  00218          .  .  .  .  .  .  .  .  X(NPOINT) = FIRSTX
  00219          .  .  .  .  .  .  .  .  Y(NPOINT) = FIRSTY
  00220          .  .  .  .  .  .  .  ...FIN
  00221          .  .  .  .  .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:39:40  PAGE  00005
  00222          .  .  .  .  .  ...FIN
  00223          .  .  .  .  .  IF (.NOT.CLOSED)
  00224          .  .  .  .  .  .  NLINE = NLINE + 1
  00225          .  .  .  .  .  .  DECODE (NCHAR,225,KEYBUF) VALUE(NLINE)
  00226   225    .  .  .  .  .  .  FORMAT (F12.0)
  00227          .  .  .  .  .  .  CALL STRTSB (MESS5)
  00228          .  .  .  .  .  .  JJ = 0
  00229          .  .  .  .  .  ...FIN
  00230          .  .  .  .  ...FIN
  00231   C      .  .  .  .
  00232   C  *** GREEN -- DELETE LAST POINT ***
  00233   C      .  .  .  .
  00234          .  .  .  .  (4)
  00235          .  .  .  .  .  DELETE-POINT-AND-REBUILD-DISPLAY
  00236          .  .  .  .  .  IF (JJ.EQ.0)
  00237          .  .  .  .  .  .  INPUT-CONTOUR-VALUE
  00238          .  .  .  .  .  ...FIN
  00239          .  .  .  .  ...FIN
  00240          .  .  .  ...FIN
  00241          .  .  .  CALL RXLOFF
  00242          .  .  ...FIN
  00243          .  ...FIN
  00244          ...FIN
  00245   C
  00246   C**************************************************************
  00247   C
  00248   C  *******  BEGIN INTERPOLATION  *******
  00249   C
  00250   C**************************************************************
  00251   C
  00252          CALL STRTSB (MESS6)
  00253          FIND-MIN-AND-MAX-X-AND-Y-AND-CONVERT-TO-NODES
  00254   C
  00255   C SET LAST POINT IN ARRAY EQUAL TO FIRST FOR WITHIN CHECK
  00256   C
  00257          NPOINT = NPOINT + 1
  00258          X(NPOINT) = X(1)
  00259          Y(NPOINT) = Y(1)
  00260   C
  00261   C CALCULATE MAP TO DIGITIZE UNITS TO SCALE FACTOR
  00262   C
  00263          FACTR = DZMAX / MAXMAP
  00264   C
  00265   C CHECK FOR NODES IN REGION
  00266   C
  00267          DO (J=MINY,MAXY)
  00268          .  SECNEW =(J-1)*NSX
  00269          .  IF(J.GE.1.AND.J.LE.NODESY)
  00270          .  .  YNODE = FLOAT(J-1) * YINC * FACTR
  00271   C      .  .
  00272          .  .  DO (I=MINX,MAXX)
  00273          .  .  .  IF(I.GE.1.AND.I.LE.NODESX)
  00274          .  .  .  .  XNODE = FLOAT(I-1) * XINC * FACTR
  00275          .  .  .  .  CALL WITHIN (NPOINT,X,Y,XNODE,YNODE,IC)
  00276          .  .  .  .  IF (IC.NE.0)
  00277   567    .  .  .  .  .  FORMAT(2F12.5)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00006
  00278          .  .  .  .  .  .  IXX = XNODE * DSMAX / DZMAX
  00279          .  .  .  .  .  .  IYY = YNODE * DSMAX / DZMAX
  00280          .  .  .  .  .  .  CALL CHGGE (1,IXX,IYY,0)
  00281          .  .  .  .  .  .  CALL STRTSB (CROSS)
  00282    C     .  .  .  .  .  .
  00283    C  CHECK FOR DATA FILE
  00284          .  .  .  .  .  .  IF (SECNEW .NE. SECOLD)
  00285    C  WRITE LAST LINE
  00286          .  .  .  .  .  .  .  IF (SECOLD .NE. -1000.) CALL DPW (IPNT1,SECOLD,SVAL,TOTX)
  00287          .  .  .  .  .  .  .  SECOLD = SECNEW
  00288    C  READ NEW LINES
  00289          .  .  .  .  .  .  .  CALL DPR (IPNT1,SECNEW,SVAL,TOTX)
  00290          .  .  .  .  .  .  ...FIN
  00291    C     .  .  .  .  .  .
  00292    C ******* FIND CLOSEST DIGITIZED POINT ON EACH LINE *******
  00293    C     .  .  .  .  .  .
  00294          .  .  .  .  .  .  KP = 0
  00295    C     .  .  .  .  .  .
  00296          .  .  .  .  .  .  DO (K=1,NLINE)
  00297          .  .  .  .  .  .  .  NPT = NCOUNT(K)
  00298          .  .  .  .  .  .  .  D = DZMAX ** 2.
  00299    C     .  .  .  .  .  .  .
  00300          .  .  .  .  .  .  .  DO (KK=1,NPT)
  00301          .  .  .  .  .  .  .  .  KP = KP + 1
  00302          .  .  .  .  .  .  .  .  D1 = (XNODE - X(KP)) **2. + (YNODE - Y(KP)) ** 2.
  00303    C     .  .  .  .  .  .  .  .
  00304          .  .  .  .  .  .  .  .  IF (D1.LE.D)
  00305          .  .  .  .  .  .  .  .  .  ITYPE = 0
  00306          .  .  .  .  .  .  .  .  .  IF (KK.EQ.1) ITYPE = 1
  00307          .  .  .  .  .  .  .  .  .  IF (KK.EQ.NPT) ITYPE = 2
  00308          .  .  .  .  .  .  .  .  .  D      = D1
  00309          .  .  .  .  .  .  .  .  .  KPOINT = KP
  00310          .  .  .  .  .  .  .  .  ...FIN
  00311          .  .  .  .  .  .  .  ...FIN
  00312    C     .  .  .  .  .  .  .
  00313          .  .  .  .  .  .  .  D1 = SQRT(D1)
  00314          .  .  .  .  .  .  .  IF (CIRCLE .AND. ITYPE.EQ.2) ITYPE = 3
  00315    C     .  .  .  .  .  .  .
  00316    C  *** FIND PERPENDICULAR DISTANCE TO LINE ***
  00317    C     .  .  .  .  .  .  .
  00318          .  .  .  .  .  .  .  WHEN (NPT.EQ.1) DIST(K) = D1
  00319          .  .  .  .  .  .  .  ELSE
  00320    C     .  .  .  .  .  .  .
  00321    C    ITYPE = 0 CLOSEST POINT ON LINE IS NEITHER THE FIRST NOR THE LAST PO
  00322    C    ITYPE = 1 -- FIRST POINT IS CLOSEST
  00323    C    ITYPE = 2 -- LAST POINT IS CLOSEST
  00324    C    ITYPE = 3 -- LAST POINT ON CIRCULAR CONTOUR IS CLOSEST
  00325    C     .  .  .  .  .  .  .  .
  00326          .  .  .  .  .  .  .  .  BIGD1 = DZMAX ** 2.
  00327          .  .  .  .  .  .  .  .  BIGD3 = DZMAX ** 2.
  00328          .  .  .  .  .  .  .  .  IF (ITYPE.NE.1)
  00329          .  .  .  .  .  .  .  .  .  GET-LINE-SEGMENT-ONE
  00330          .  .  .  .  .  .  .  .  .  FIND-DISTANCE-PARAMETERS
  00331          .  .  .  .  .  .  .  .  .  BIGD2 = SQRT(L2SQ)
  00332    C     .  .  .  .  .  .  .  .  .
  00333          .  .  .  .  .  .  .  .  .  WHEN (SD1.GT.SMALLD) BIGD1 = BIGD2
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00007
  00334   C      .  .  .  .  .  .  .  .
  00335          .  .  .  .  .  .  .  .  ELSE
  00336          .  .  .  .  .  .  .  .  .  BIGD1 = SQRT(L1SQ - SD1 ** 2.)
  00337          .  .  .  .  .  .  .  .  ...FIN
  00338          .  .  .  .  .  .  .  ...FIN
  00339   C      .  .  .  .  .  .  .  .
  00340          .  .  .  .  .  .  .  .  IF (ITYPE.NE.2)
  00341   C      .  .  .  .  .  .  .  .
  00342          .  .  .  .  .  .  .  .  IF (ITYPE.EQ.3) KPOINT = KPOINT - NCOUNT(K) + 1
  00343          .  .  .  .  .  .  .  .  GET-LINE-SEGMENT-TWO
  00344          .  .  .  .  .  .  .  .  FIND-DISTANCE-PARAMETERS
  00345          .  .  .  .  .  .  .  .  BIGD2 = SQRT(L1SQ)
  00346   C      .  .  .  .  .  .  .  .
  00347          .  .  .  .  .  .  .  .  WHEN (SD1.LT.0) BIGD3 = BIGD2
  00348   C      .  .  .  .  .  .  .  .
  00349          .  .  .  .  .  .  .  .  ELSE
  00350          .  .  .  .  .  .  .  .  .  BIGD3 = SQRT(L1SQ - SD1 ** 2.)
  00351          .  .  .  .  .  .  .  .  ...FIN
  00352          .  .  .  .  .  .  .  ...FIN
  00353   C      .  .  .  .  .  .  .  .
  00354          .  .  .  .  .  .  .  .  DIST(K) = AMIN1(BIGD1,BIGD2,BIGD3)
  00355          .  .  .  .  .  .  .  ...FIN
  00356          .  .  .  .  .  .  ...FIN
  00357   C      .  .  .  .  .  .
  00358   C  **** CALCULATE NODAL VALUE  ****
  00359   C      .  .  .  .  .
  00360          .  .  .  .  .  BOT = 0.0
  00361          .  .  .  .  .  TOP = 0.0
  00362   C      .  .  .  .  .
  00363          .  .  .  .  .  DO (K=1,NLINE)
  00364          .  .  .  .  .  .  IF (DIST(K) .EQ. 0)
  00365          .  .  .  .  .  .  .  SVAL(I) = VALUE(K)
  00366          .  .  .  .  .  .  .  GO TO 68
  00367          .  .  .  .  .  .  ...FIN
  00368          .  .  .  .  .  .  TOP = TOP + VALUE(K) / DIST(K)
  00369          .  .  .  .  .  .  BOT = BOT + 1. / DIST(K)
  00370          .  .  .  .  .  ...FIN
  00371   C      .  .  .  .  .
  00372          .  .  .  .  .  SVAL(I) = TOP / BOT
  00373   68     .  .  .  .  .  CONTINUE
  00374          .  .  .  .  ...FIN
  00375          .  .  .  ...FIN
  00376          .  .  ...FIN
  00377          .  ...FIN
  00378          ...FIN
  00379   C
  00380   C  *** STORE LAST LINE
  00381          CALL UPW (IPNT1,SECOLD,SVAL,TOTX)
  00382   C
  00383   C*************************************************************
  00384   C
  00385   C  **********  INTERPOLATION COMPLETE  **********
  00386   C
  00387   C*************************************************************
  00388   C
  00389          CALL STOPSR (MES96,CROSS)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00008
  00390    C
  00391           NPOINT = NPOINT - 1
  00392    C
  00393           CALL STRTSB (MESS7)
  00394           CALL BxKEY (KEYBUF,30,ICC,NCHAR)
  00395           DECODE (NCHAR,270,KEYBUF) NOPT
  00396    270    FORMAT (I1)
  00397           CALL STOPSB (MESS7)
  00398           CALL SUBDF (LINES,900,2)
  00399           LMN = 0
  00400    C   *** START NEW SEQUENCE  ***
  00401    C
  00402    C
  00403    C    CONTROL OPTIONS:
  00404    C      NOPT=0 --- TERMINATE SEQUENCE
  00405    C      NOPT=1 --- START NEW SEQUENCE
  00406    C      NOPT=2 --- SAVE LAST LINE AND CONTINUE SEQUENCE
  00407           SELECT (NOPT)
  00408           .   (0)
  00409           .   .   CLOSED = .FALSE.
  00410           .   ...FIN
  00411    C      .
  00412           .   (1)
  00413           .   .   START-NEW-SEQUENCE
  00414           .   ...FIN
  00415    C      .
  00416           .   (2)
  00417           .   .   NSTART = NPOINT - NCOUNT(NLINE) + 1
  00418           .   .   JJ = 0
  00419    C      .   .
  00420           .   .   DO (KK=NSTART,NPOINT)
  00421           .   .   .   JJ = JJ + 1
  00422           .   .   .   X(JJ) = X(KK)
  00423           .   .   .   Y(JJ) = Y(KK)
  00424           .   .   .   IDSX = X(JJ) * DSMAX / DZMAX
  00425           .   .   .   IDSY = Y(JJ) * DSMAX / DZMAX
  00426           .   .   .   WHEN (JJ.EQ.1) DISPLAY-LINE-VALUE
  00427    C      .   .   .
  00428           .   .   .   ELSE
  00429           .   .   .   .   CALL AVEC (IDSX,IDSY)
  00430           .   .   .   ...FIN
  00431           .   .   ...FIN
  00432    C      .   .
  00433           .   .   NCOUNT(1) = NCOUNT(NLINE)
  00434           .   .   NPOINT    = NCOUNT(NLINE)
  00435           .   .   VALUE(1)  = VALUE(NLINE)
  00436           .   .   NLINE     = 2
  00437           .   .   INPUT-CONTOUR-VALUE
  00438           .   .   JJ = 0
  00439           .   ...FIN
  00440           ...FIN
  00441    C
  00442           GO TO 10
  00443    99     WRITE (5,599)
  00444    599    FORMAT (' OUTPUT DATA FILE NOT FOUND')
  00445    999    TASK = RAD50('APPLIC')
```

```
00446          CALL REQUES (TASK,,IDS)
00447          CALL EXIT
00448    C
00449    C
         ---------------------------------------

00450          TO START-NEW-SEQUENCE
00451          . CALL STRTSB (MESS3)
00452          . CALL BXKEY (KEYBUF,30,ICC,NCHAR)
00453          . DECODE (NCHAR,224,KEYBUF) OCODE
00454          . CALL STOPSB (MESS3)
00455          . NORMAL = .FALSE.
00456          . CIRCLE = .FALSE.
00457          . RIVER  = .FALSE.
00458          . SELECT (OCODE)
00459          . .  ('N')  NORMAL = .TRUE.
00460          . .  ('C')  CIRCLE = .TRUE.
00461          . .  ('R')  RIVER  = .TRUE.
00462          . ...FIN
00463          . NPOINT = 0
00464          . NLINE = 1
00465          . INPUT-CONTOUR-VALUE
00466          . JJ=0
00467          ...FIN
         ---------------------------------------

00468          TO INPUT-CONTOUR-VALUE
00469          . CLOSED=.FALSE.
00470          . CALL STRTSB(MESS4)
00471          . CALL BXKEY(KEYBUF,30,ICC,NCHAR)
00472          . DECODE(NCHAR,225,KEYBUF) VALUE(NLINE)
00473          . CALL STOPSB(MESS4)
00474          . CALL STRTSB(MESS5)
00475          ...FIN
         ---------------------------------------

00476          TO STORE-POINTS-AND-CALCULATE-GRAPHIC-VALUES
00477    C     .
00478    C     . IF (DZX.LT.0.) DZX = 0.
00479    C     . IF (DZX.GT.DZXMAX) DZX = DZXMAX
00480    C     . IF (DZY.LT.0.) DZY = 0.
00481    C     . IF (DZY.GT.DZYMAX) DZY = DZYMAX
00482          . IDSX = DZX * DSMAX / DZMAX
00483          . IDSY = DZY * DSMAX / DZMAX
00484    0     . WRITE(5,5443) IDSX,IDSY
00485    5443  . FORMAT(' IDSX,IDSY=',2I5)
00486    0     . WRITE (5,5114) SCALE,DZMAX,DZX,DZY
00487          . X(NPOINT) = DZX
00488          . Y(NPOINT) = DZY
00489          ...FIN
00490    C
00491    C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00010
        ----------------------------------------

 00492          TO DISPLAY-LINE-VALUE
 00493          .  CALL INTENS (3)
 00494          .  CALL CHRTYP (0,0,0)
 00495          .  CALL CHRSCL (0)
 00496          .  CALL APOS (IDSX,IDSY)
 00497   D      .  WRITE(5,5444) NLINE,VALUE(NLINE),IDSX,IDSY,DSMAX
 00498   5444   .  FORMAT(I4,F8.1,2I5,F10.2)
 00499   D      .  WRITE(5,5445) DZMAX,SCALE,XMAX,YMAX,MAXMAP
 00500   5445   .  FORMAT(5F12.2)
 00501          .  WHEN (VALUE(NLINE).LT.100000.)
 00502          .  .  CALL FNMBR (VALUE(NLINE),'(F10.3)')
 00503          .  ...FIN
 00504   C      .
 00505          .  ELSE
 00506          .  .  CALL FNMBR (VALUE(NLINE),'(1PE13.5)')
 00507          .  ...FIN
 00508   C      .
 00509          .  CALL INTENS (7)
 00510          .  CALL APOS (IDSX,IDSY)
 00511          ...FIN
 00512   C

        ----------------------------------------

 00513          TO DELETE-POINT-AND-REBUILD-DISPLAY
 00514          .  CALL ENDSB
 00515          .  CALL SUBDF (LINES,900,2)
 00516          .  IF (JJ.NE.0)
 00517          .  .  JJ = JJ - 1
 00518          .  .  NPOINT = NPOINT - 1
 00519          .  .  NCOUNT(NLINE) = JJ
 00520          .  ...FIN
 00521          .  DISPLAY-NUMBER-OF-POINTS-USED
 00522          .  IF (JJ.EQ.0) NLINE = NLINE - 1
 00523          .  IF (NLINE.NE.0)
 00524          .  .  KK = 0
 00525   C      .  .
 00526          .  .  DO (L=1,NLINE)
 00527   C      .  .  .
 00528          .  .  .  DO (K=1,NCOUNT(L))
 00529          .  .  .  .  KK = KK + 1
 00530          .  .  .  .  IDSX = X(KK) * DSMAX / DZMAX
 00531          .  .  .  .  IDSY = Y(KK) * DSMAX / DZMAX
 00532          .  .  .  .  WHEN (K.EQ.1) DISPLAY-LINE-VALUE
 00533   C      .  .  .  .
 00534          .  .  .  .  ELSE
 00535          .  .  .  .  .  CALL AVEC (IDSX,IDSY)
 00536          .  .  .  .  ...FIN
 00537          .  .  .  ...FIN
 00538          .  .  ...FIN
 00539          .  ...FIN
 00540          ...FIN
 00541   C
```

```
00542          TO DISPLAY-NUMBER-OF-POINTS-USED
00543          .  CALL SUBDF(NPNT,30,LMN)
00544          .  LMN = 2
00545          .  CALL AREA(1)
00546          .  CALL INTENS(7)
00547          .  CALL CHRTYP(0,0,0)
00548          .  CALL CHRSCL(0)
00549          .  CALL APOS(124,6)
00550          .  CALL INMBR(NPOINT,'(I4)')
00551          .  CALL AREA(0)
00552          .  CALL ENDSB
00553          ...FIN

          ----------------------------------------

00554          TO FIND-MIN-AND-MAX-X-AND-Y-AND-CONVERT-TO-NODES
00555    C     .
00556    C DETERMINE MINIMUM AND MAXIMUM X AND Y VALUES FROM ARRAY
00557    C     .
00558          .  XXMIN = X(1)
00559          .  XXMAX = X(1)
00560          .  YYMIN = Y(1)
00561          .  YYMAX = Y(1)
00562    C     .
00563          .  DO (J=2,NPOINT)
00564          .  .  IF (X(J).LT.XXMIN) XXMIN = X(J)
00565          .  .  IF (X(J).GT.XXMAX) XXMAX = X(J)
00566          .  .  IF (Y(J).LT.YYMIN) YYMIN = Y(J)
00567          .  .  IF (Y(J).GT.YYMAX) YYMAX = Y(J)
00568          .  ...FIN
00569    C     .
00570    C CONVERT DIGITIZER UNITS TO NODES
00571    C     .
00572          .  FACT = MAXMAP / DZMAX
00573          .  MINX = (XXMIN * FACT) / XINC + 1
00574          .  MAXX = (XXMAX * FACT) / XINC + 2
00575          .  MINY = (YYMIN * FACT) / YINC + 1
00576          .  MAXY = (YYMAX * FACT) / YINC + 2
00577          ...FIN
00578    C

          ----------------------------------------

00579          TO GET-LINE-SEGMENT-ONE
00580    C     .
00581    C  USE CLOSEST POINT AND LAST POINT ON LINE
00582    C     .
00583          .  X0 = X(KPOINT - 1)
00584          .  Y0 = Y(KPOINT - 1)
00585          .  X1 = X(KPOINT)
00586          .  Y1 = Y(KPOINT)
00587          ...FIN
00588    C
```

```
          ----------------------------------------

00589           TO GET-LINE-SEGMENT-TWO
00590     C     .
00591     C  USE CLOSEST POINT AND NEXT POINT ON LINE
00592     C     .
00593           .  X0 = X(KPOINT)
00594           .  Y0 = Y(KPOINT)
00595           .  X1 = X(KPOINT + 1)
00596           .  Y1 = Y(KPOINT + 1)
00597           ...FIN
00598     C

          ----------------------------------------

00599           TO FIND-DISTANCE-PARAMETERS
00600           .  FX1 = X0 - X1
00601           .  FY1 = Y0 - Y1
00602           .  FX2 = X0 - XNODE
00603           .  FY2 = Y0 - YNODE
00604           .  FX3 = X1 - XNODE
00605           .  FY3 = Y1 - YNODE
00606           .  SMALLD = SQRT(FX1 * FX1  +  FY1 * FY1)
00607           .  L1SQ = FX2 * FX2  + FY2 * FY2
00608           .  L2SQ = FX3 * FX3  + FY3 * FY3
00609           .  SD1 = SMALLD / 2. + (L1SQ - L2SQ) / (2. * SMALLD)
00610           ...FIN

          ----------------------------------------

00611           TO SET-UP-SUBDISPLAY-FILES
00612     C     .
00613     C*************************************************************
00614     C     .
00615     C     .
00616     C **********  MAJOR PROGRAM SUBDISPLAYS **********
00617     C     .
00618     C     .
00619     C*************************************************************
00620     C     .
00621     C     .
00622           .  CALL SUBDF (LABEL,520,1)
00623           .  CALL INTENS (7)
00624           .  CALL CHRTYP (1,0,0)
00625           .  CALL CHRSCL (3)
00626           .  CALL APOS (300,1000)
00627           .  CALL TEXT ('SURFACE GENERATOR',0)
00628           .  CALL AREA(1)
00629           .  CALL CHRSCL (0)
00630           .  CALL APOS (0,30)
00631           .  CALL TEXT ('POINTS AVAILABLE = 400',-24)
00632           .  CALL TEXT ('POINTS USED      =',0)
00633           .  CALL CHRSCL (1)
00634           .  CALL APOS (70,950)
00635           .  CALL TEXT ('CONTROLS')
00636           .  IX = 0
```

```
(FLECS VERSION 22,46)     14-SEP-79   08139140  PAGE  00014
  00693   C*****************************************************************
  00694   C
  00695           .  CALL SUBOF (MES81,55,1)
  00696           .  CALL INTENS (7)
  00697           .  CALL CHRSCL (1)
  00698           .  CALL BLINK  (1)
  00699           .  CALL APOS   (125,500)
  00700           .  CALL TEXT   ('*')
  00701           .  CALL BLINK  (0)
  00702           .  CALL APOS   (150,500)
  00703           .  CALL TEXT('DEPRESS RED BUTTON AT (0,0) ON MAP',-30)
  00704           .  CALL TEXT('(THIS IS ALSO START OF MAP SCALE)')
  00705           .  CALL ENDSB(ISIZ(3))
  00706   C
  00707           .  CALL SUBOF (MES82,50,1)
  00708           .  CALL INTENS (7)
  00709           .  CALL CHRSCL (1)
  00710           .  CALL BLINK  (1)
  00711           .  CALL APOS   (75,500)
  00712           .  CALL TEXT   ('*')
  00713           .  CALL BLINK  (0)
  00714           .  CALL APOS   (100,500)
  00715           .  CALL TEXT('DEPRESS WHITE BUTTON AT END OF SPECIFIED',0)
  00716           .  CALL TEXT(' HORIZONTAL SCALE ON MAP')
  00717           .  CALL ENDSB(ISIZ(4))
  00718   C
  00719           .  CALL SUBOF (MES821,45,1)
  00720           .  CALL INTENS (7)
  00721           .  CALL CHRSCL (1)
  00722           .  CALL BLINK  (1)
  00723           .  CALL APOS   (175,970)
  00724           .  CALL TEXT   ('*')
  00725           .  CALL BLINK  (0)
  00726           .  CALL APOS   (200,970)
  00727           .  CALL TEXT('ENTER MAP SCALE DISTANCE ON DIGITIZER KEYBOARD',0)
  00728           .  CALL TEXT('  (F10,0)')
  00729           .  CALL ENDSB(ISIZ(5))
  00730   C
  00731           .  CALL SUBOF (MES83,45,1)
  00732           .  CALL INTENS (3)
  00733           .  CALL CHRTYP (0,0,0)
  00734           .  CALL CHRSCL (1)
  00735           .  CALL BLINK (1)
  00736           .  CALL APOS   (175,970)
  00737           .  CALL TEXT   ('*')
  00738           .  CALL BLINK  (0)
  00739           .  CALL APOS   (200,970)
  00740           .  CALL TEXT (' ENTER LINE TYPE ON DIGITIZER KEYBOARD((N,C OR R)')
  00741           .  CALL ENDSB(ISIZ(6))
  00742   C
  00743           .  CALL SUBOF (MES84,45,1)
  00744           .  CALL INTENS (3)
  00745           .  CALL CHRTYP (0,0,0)
  00746           .  CALL CHRSCL (1)
  00747           .  CALL BLINK (1)
  00748           .  CALL APOS   (175,970)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:39:40  PAGE  00013
  00637           .  IY = 943
  00638           .  CALL APOS (IX,IY)
  00639           .  CALL RVEC (220,0)
  00640           .  CALL RVEC (0,30)
  00641           .  CALL RVEC (-220,0)
  00642           .  CALL RVEC (0,-30)
  00643           .  CALL RPOS (2,2)
  00644           .  CALL RVEC (216,0)
  00645           .  CALL RVEC (0,26)
  00646           .  CALL RVEC (-216,0)
  00647           .  CALL RVEC (0,-26)
  00648           .  CALL INTENS (5)
  00649           .  CALL CHRSCL (0)
  00650           .  CALL APOS (30,900)
  00651           .  CALL TEXT ('V360 DISPLAY',-3)
  00652           .  CALL RVEC (82,0)
  00653           .  CALL APOS (14,860)
  00654           .  CALL TEXT ('LIGHT PEN OPTIONS:')
  00655           .  CALL APOS (45,500)
  00656           .  CALL TEXT ('DIGITIZER',-3)
  00657           .  CALL RVEC (60,0)
  00658           .  CALL APOS (0,470)
  00659           .  CALL TEXT ('CURSOR BUTTONS:',-20)
  00660           .  CALL TEXT ('       RED --- INITIALIZE REGION',-20)
  00661           .  CALL TEXT ('       WHITE - DIGITIZE POINT',-20)
  00662           .  CALL TEXT ('       BLUE -- TERMINATE LINE',-20)
  00663           .  CALL TEXT ('       GREEN - DELETE LAST POINT',-20)
  00664           .  CALL TEXT ('       YELLOW- HOLD POSITION ON MAP',-40)
  00665           .  CALL TEXT ('KEYBOARD LINE OPTIONS:',-20)
  00666           .  CALL TEXT ('       N---NORMAL CONTOURS:',-20)
  00667           .  CALL TEXT ('       C---CIRCULAR CONTOURS:',-20)
  00668           .  CALL TEXT ('       R---RIVER BOUNDARY:',-40)
  00669     C     .
  00670           .  CALL TEXT ('KEYBOARD CONTROL OPTIONS:',-20)
  00671           .  CALL TEXT ('       0 --- TERMINATE SEQUENCE',-20)
  00672           .  CALL TEXT ('       1 --- START NEW SEQUENCE',-20)
  00673           .  CALL TEXT ('       2 --- SAVE LAST LINE',-40)
  00674           .  CALL TEXT ('OTHER KEYBOARD INPUT',-20)
  00675           .  CALL TEXT ('AS REQUESTED:')
  00676     C     .
  00677     C  *** LIGHT BUTTONS ***
  00678     C     .
  00679           .  CALL MENU (830,40,1,'         RETURN TO CONTROL',
  00680         1.   '         WRITE BINARY FILE',
  00681         1.   '         SETUP CONTOURS','         EDIT MATRIX',
  00682         2.   '         CONTOURS (ON/OFF)','         ABORT SEQUENCE',
  00683         3.   '         START SEQUENCE')
  00684           .  CALL AREA(0)
  00685           .  CALL ENDSR(ISIZ(1))
  00686           .  CALL STRTSB (LABEL)
  00687     C     .
  00688     C     .
  00689     C*************************************************************
  00690     C     .
  00691     C  ******* DIGITIZER SUBDISPLAY FILE *******
  00692     C     .
```

```
00749          .  CALL TEXT   ('*')
00750          .  CALL BLINK  (0)
00751          .  CALL APOS   (200,970)
00752          .  CALL TEXT ('INPUT CONTOUR VALUE ON DIGITIZER ',0)
00753          .  CALL TEXT ('KEYBOARD (F12,0)')
00754          .  CALL ENDSB(ISIZ(7))
00755    C     .
00756          .  CALL SUBDF (MESS41,30,1)
00757          .  CALL INTENS (3)
00758          .  CALL CHRTYP (0,0,0)
00759          .  CALL CHRSCL (1)
00760          .  CALL APOS   (200,950)
00761          .  CALL TEXT (' OR "DONE" TO CLOSE REGION')
00762          .  CALL ENDSB(ISIZ(8))
00763    C     .
00764          .  CALL SUBDF (MESS5,35,1)
00765          .  CALL INTENS (3)
00766          .  CALL CHRTYP (0,0,0)
00767          .  CALL CHRSCL (1)
00768          .  CALL BLINK (1)
00769          .  CALL APOS   (175,970)
00770          .  CALL TEXT   ('*')
00771          .  CALL BLINK  (0)
00772          .  CALL APOS   (200,970)
00773          .  CALL TEXT ('BEGIN DIGITIZING (WHITE BUTTON)')
00774          .  CALL ENDSB(ISIZ(9))
00775    C     .
00776          .  CALL SUBDF (MESS6,25,1)
00777          .  CALL INTENS (3)
00778          .  CALL CHRTYP (0,0,0)
00779          .  CALL CHRSCL (3)
00780          .  CALL BLINK (1)
00781          .  CALL APOS (200,500)
00782          .  CALL TEXT (' INTERPOLATING')
00783          .  CALL ENDSB(ISIZ(11))
00784    C     .
00785          .  CALL SUBDF (MESS7,60,1)
00786          .  CALL INTENS (3)
00787          .  CALL CHRTYP (0,0,0)
00788          .  CALL CHRSCL (1)
00789          .  CALL BLINK (1)
00790          .  CALL APOS   (175,970)
00791          .  CALL TEXT   ('*')
00792          .  CALL BLINK  (0)
00793          .  CALL APOS   (200,970)
00794          .  CALL TEXT ('INTERPOLATION COMPLETE',-30)
00795          .  CALL TEXT ('ENTER CONTROL OPT. ON DIGITIZER KEYBOARD',0)
00796          .  CALL TEXT (' ( 0,1 OR 2)')
00797          .  CALL ENDSB(ISIZ(12))
00798    C     .
00799          .  CALL SUBDF (CROSS,15,1)
00800          .  CALL INTENS (5)
00801          .  CALL CHRSCL (0)
00802          .  CALL POINTR (1)
00803          .  CALL APOS (0,0)
00804          .  CALL RPOS (-4,-4)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08139140  PAGE  00016
  00805           .  CALL TEXT ('+')
  00806           .  CALL ENDSR(ISIZ(13))
  00807   0       .  WRITE(5,5999) ISIZ
  00808   5999    .  FORMAT(' ISIZ= ',13I4)
  00809           ...FIN
  00810           END

          ----------------------------------------

                PROCEDURE CROSS-REFERENCE TABLE

          00513 DELETE-POINT-AND-REBUILD-DISPLAY
                00235

          00492 DISPLAY-LINE-VALUE
                00195  00426  00532

          00542 DISPLAY-NUMBER-OF-POINTS-USED
                00188  00521

          00599 FIND-DISTANCE-PARAMETERS
                00330  00344

          00554 FIND-MIN-AND-MAX-X-AND-Y-AND-CONVERT-TO-NODES
                00253

          00579 GET-LINE-SEGMENT-ONE
                00329

          00589 GET-LINE-SEGMENT-TWO
                00343

          00468 INPUT-CONTOUR-VALUE
                00237  00437  00465

          00611 SET-UP-SUBDISPLAY-FILES
                00055

          00450 START-NEW-SEQUENCE
                00133  00165  00413

          00476 STORE-POINTS-AND-CALCULATE-GRAPHIC-VALUES
                00189


          (FLECS VERSION 22.46)

          ----------------------------------------
```

CONTUR.FLX

SURFACE CONTOURING

```
      ----------------------------------------

00001 C ********  [351,101]CONTUR,FLX  *******
00002 C
00003 C
00004       BYTE FNAME,SRCMAP,DSPMAP
00005       BYTE KEYBUF(80)
00006       INTEGER OVRFLW
00007       LOGICAL DONE,ENDDSP
00008 C
00009       REAL MAXMAP
00010 C
00011       DIMENSION LAST(100),P1(256),P2(256),NDIGIT(72)
00012 C
00013       COMMON MAP(3000)
00014       COMMON/DSP/ MAIN(100),KEYOUT(100),LINDSP(4000),BKGRND(50),
00015      1             LABEL(100),LPLAB(200),MESS1(60),NAMDSP(20),
00016      2             OVRFLW(100)
00017       COMMON/TND/ INITDZ,DZXMAX,DZYMAX,DZMAX,NODESX,NODESY,XINC,YINC,
00018      1            NSX,TOTSEC,TOTX,FNAME(30),SRCMAP(30),DSPMAP(30),
00019      2            XMAX,YMAX,MAXMAP,DSMAX
00020 C
00021       EQUIVALENCE (NDIGIT(1),INITDZ)
00022 C*****************************************************************
00023 C*
00024 C  ********  INITIALIZE DISPLAY   ********
00025 C
00026 C*************************************************************
00027 C
00028       CALL MAINDF (MAIN,100,0,1,0,2)
00029       CALL CLKOFF
00030 C
00031       SETUP-DISPLAY-FILES
00032 C
00033 C
00034 C*************************************************************
00035 C
00036 C
00037 C
00038 C
00039 C*************************************************************
00040 C
00041 C  ***************  START PROGRAM  ***************
00042 C
00043 C*************************************************************
00044 C
00045 C
00046       CALL DPFILE(4,'SURFSPC',1,,IPNT0)
00047       CALL DPR (IPNT0,0,,NDIGIT,72,)
00048       CALL DPFILE(4,'SURFTMP',TOTSEC,IPNT1)
00049 C  DISPLAY MAP BOUNDARY
00050       IXM=XMAX*DSMAX/MAXMAP
00051       IYM=YMAX*DSMAX/MAXMAP
00052       WHEN (SRCMAP(1).NE.'N')
00053       .  CALL RESSDF (3,DSPMAP,MAP)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08140141  PAGE  00002
  00054          .  CALL CALLSB (MAP)
  00055          ...FIN
  00056          ELSE
  00057          .  CALL SUBDF(MAP,3000,0)
  00058          .  CALL INTENS(3)
  00059          .  CALL APOS(0,0)
  00060          .  CALL AVEC(IXM,0)
  00061          .  CALL AVEC(IXM,IYM)
  00062          .  CALL AVEC(0,IYM)
  00063          .  CALL AVEC(0,0)
  00064          .  CALL ENDSB
  00065          ...FIN
  00066    C
  00067          NARS = 0
  00068          NARG = IARG
  00069          IARG = 11
  00070    C
  00071    C
  00072    C*****************************************************************
  00073    C
  00074    C      BEGIN CONTOURING
  00075    C
  00076    C*****************************************************************
  00077    C
  00078    C
  00079          II1 = 0
  00080          CALL SUBDF(LINDSP,4000,II1)
  00081          II1 = 2
  00082          CALL CHRSCL (0)
  00083          CALL INTENS (7)
  00084          CALL ENDSB
  00085          ENDDSP = .FALSE.
  00086          II2 = 1
  00087          IKB = 0
  00088    19    CALL GRATTN (0,ITYP,'LP','KB')
  00089          SELECT (ITYP)
  00090    C ** KEYBOARD INTERRUPT **
  00091          .  ('KB')
  00092          .  .  IF (.NOT.ENDDSP)
  00093          .  .  .  CALL STOPSB (NAMDSP)
  00094          .  .  .  DO (J=1,12) KEYBUF(J) =  ' '
  00095          .  .  .  CALL GRECHO (KEYBUF,NCHAR,210,940,1,1,IKB)
  00096          .  .  .  DECODE (NCHAR,220,KEYBUF) VAL
  00097    220   .  .  .  FORMAT (F10.0)
  00098          .  .  .  CALL SUBOF (NAMDSP,20,II2)
  00099          .  .  .  II2 = 3
  00100          .  .  .  CALL INTENS (4)
  00101          .  .  .  CALL CHRSCL (1)
  00102          .  .  .  CALL APOS (210,940)
  00103          .  .  .  CALL TEXT (KEYBUF,0,12)
  00104          .  .  .  CALL ENDSB
  00105          .  .  .  CALL STOPSB (KEYOUT)
  00106          .  .  .  CALL STRTSB (NAMDSP)
  00107          .  .  .  DISPLAY-CONTOUR-LINE
  00108          .  .  ...FIN
  00109          .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08140141  PAGE  00003
 00110   C ** LIGHT PEN INTERRUPTS **
 00111           .  ('LP')
 00112           .  .  CALL LTPEN (IO,ITIP)
 00113           .  .  IF (ITIP.EQ.1)
 00114           .  .  .  SELECT (IO)
 00115           .  .  .  .  (1)
 00116           .  .  .  .  .  CALL CLRDEV ('KB')
 00117           .  .  .  .  .  CALL WAIT (2,2,MMM)
 00118           .  .  .  .  .  TASK = RAD50 ('APPLIC')
 00119           .  .  .  .  .  CALL REQUES (TASK,,IDS)
 00120           .  .  .  .  .  IF (IO.EQ.1) CALL EXIT
 00121           .  .  .  .  ...FIN
 00122           .  .  .  .  (2)
 00123           .  .  .  .  .  CONTINUE
 00124           .  .  .  .  ...FIN
 00125           .  .  .  .  (3)
 00126           .  .  .  .  .  CALL SUBOF (LINDSP,4000,III)
 00127           .  .  .  .  .  CALL CHRSCL (0)
 00128           .  .  .  .  .  CALL INTENS (7)
 00129           .  .  .  .  .  CALL ENDSB
 00130           .  .  .  .  .  CALL STOPSB (OVRFLW)
 00131           .  .  .  .  .  ENDDSP = .FALSE.
 00132           .  .  .  .  ...FIN
 00133           .  .  .  ...FIN
 00134           .  .  ...FIN
 00135           .  ...FIN
 00136           ...FIN
 00137           GO TO 19
 00138   C

         ----------------------------------------

 00139           TO DISPLAY-CONTOUR-LINE
 00140           .  CALL OPENSB (LINDSP,4000,III)
 00141           .  INUM = 1
 00142           .  IF(VAL.GE.10.) INUM = 2
 00143           .  IF(VAL.GE.100.) INUM = 3
 00144           .  IF(VAL.GE.10000.) INUM = 4
 00145           .  IF(VAL.LT.1.) INUM = 5
 00146   C       .
 00147           .  DO (I=1,100) LAST(I)=0
 00148   C       .
 00149   C  READ DISC BASED DATA
 00150           .  DO ( I=1,NODESY-1)
 00151           .  .  ADRS = (I-1)*NSX
 00152           .  .  CALL DPR(IPNT1,ADRS,P1,TOTX)
 00153           .  .  ADRS = ADRS+FLOAT(NSX)
 00154           .  .  CALL DPR(IPNT1,ADRS,P2,TOTX)
 00155           .  .  NLINE = 0
 00156   C       .  .
 00157   C  SCAN BOTTOM OR TOP LINE FOR INTERSECTION
 00158           .  .  DO ( MODE=1,2)
 00159           .  .  .  II = I+MODE-1
 00160           .  .  .  JJ = 0
 00161   20      .  .  .  JJ = JJ+1
 00162           .  .  .  IF (JJ.LT.NODESX)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:40:41  PAGE  00004
  00163                 .  .  .  .  J = JJ
  00164                 .  .  .  .  SELECT (MODE)
  00165                 .  .  .  .  .  (1)
  00166                 .  .  .  .  .  .  CALL INRT(P1(JJ),P1(JJ+1),VAL,INB,FRAC)
  00167   D             .  .  .  .  .  .  WRITE(5,519) JJ,P1(JJ),P1(JJ+1),VAL,INB,FRAC
  00168   519           .  .  .  .  .  .  FORMAT(I5,3F8.2,I5,F8.5)
  00169                 .  .  .  .  .  .  ...FIN
  00170                 .  .  .  .  .  (2)
  00171                 .  .  .  .  .  .  CALL INRT(P2(JJ),P2(JJ+1),VAL,INB,FRAC)
  00172                 .  .  .  .  .  .  ...FIN
  00173                 .  .  .  .  ...FIN
  00174                 .  .  .  .  IF (INB.NE.1) GO TO 20
  00175   D             .  .  .  .  WRITE(5,520) MODE,II,JJ
  00176   520           .  .  .  .  FORMAT(' MODE,II,JJ=',3I4)
  00177   C             .  .  .  .
  00178   C  *** FOUND AN INTERSECTION ON A HORIZONTAL LINE
  00179                 .  .  .  .  Y = FLOAT(II-1)*YINC
  00180                 .  .  .  .  X = (FLOAT(JJ-1) + FRAC)*XINC
  00181                 .  .  .  .  ICORN = JJ*1000 + II
  00182                 .  .  .  .  NSCAN = 1
  00183                 .  .  .  .  IF (NLINE.NE.0)
  00184   C    CHECK FOR LINE ALREADY TRACED
  00185                 .  .  .  .  .  NSCAN = 4
  00186                 .  .  .  .  .  DO (KK=1,NLINE)
  00187                 .  .  .  .  .  .  IF(ICORN.EQ.LAST(KK)) GO TO 20
  00188                 .  .  .  .  .  ...FIN
  00189                 .  .  .  .  ...FIN
  00190   C             .  .  .  .
  00191   C  START NEW LINE
  00192                 .  .  .  .  NLINE = NLINE + 1
  00193                 .  .  .  .  LAST(NLINE) = ICORN
  00194                 .  .  .  .  IX = X*DSMAX/MAXMAP + 0.5
  00195                 .  .  .  .  IY = Y*DSMAX/MAXMAP+ 0.5
  00196                 .  .  .  .  CALL APOS (IX,IY)
  00197                 .  .  .  .  IF (NABS.EQ.0.OR.NABS.EQ.25)
  00198                 .  .  .  .  .  CALL INTENS (3)
  00199                 .  .  .  .  .  CONDITIONAL
  00200                 .  .  .  .  .  .  (INUM.EQ.1) CALL FNMBR(VAL,'(F4.2)',1)
  00201                 .  .  .  .  .  .  (INUM.EQ.2) CALL FNMBR(VAL,'(F4.1)',1)
  00202                 .  .  .  .  .  .  (INUM.EQ.3) CALL FNMBR(VAL,'(F5.0)',1)
  00203                 .  .  .  .  .  .  (OTHERWISE) CALL FNMBR (VAL,'(1PE10.3)',1)
  00204                 .  .  .  .  .  ...FIN
  00205                 .  .  .  .  .  NABS = 0
  00206                 .  .  .  .  ...FIN
  00207                 .  .  .  .  CALL INTENS (7)
  00208                 .  .  .  .  NABS = NABS+1
  00209   D             .  .  .  .  WRITE(5,540) IX,IY
  00210   540           .  .  .  .  FORMAT(' APOS=',2I5)
  00211   C             .  .  .  .
  00212   C  FIND WHICH DIRECTION IT GOES---FOLLOW CONTOUR
  00213   C             .  .  .
  00214                 .  .  .  .  IF (MODE.EQ.1)
  00215                 .  .  .  .  .  CALL INBT(P2(J),P2(J+1),VAL,INB,FRAC)
  00216                 .  .  .  .  .  IF(INB.EQ.1)
  00217   C    HIT TOP LINE
  00218   40            .  .  .  .  .  .  IP = II+1-MODE
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:40:41  PAGE  0005
  00219          .  .  .  .  .  .  Y = FLOAT(IP)*YINC
  00220          .  .  .  .  .  .  X = (FLOAT(J-1)+FRAC)*XINC
  00221          .  .  .  .  .  .  ICORN = J*1000+IP+1
  00222          .  .  .  .  .  .  NSCAN = 1
  00223          .  .  .  .  .  .  IX = X*DSMAX/MAXMAP + 0.5
  00224          .  .  .  .  .  .  IY = Y*DSMAX/MAXMAP + 0.5
  00225          .  .  .  .  .  .  CALL VEC (IX,IY)
  00226    D     .  .  .  .  .  .  WRITE(5,541) NSCAN,IX,IY
  00227    541   .  .  .  .  .  .  FORMAT(3I5)
  00228          .  .  .  .  .  .  LAST(NLINE) = ICORN
  00229          .  .  .  .  .  .  GO TO 20
  00230          .  .  .  .  .  ...FIN
  00231          .  .  .  .  ...FIN
  00232          .  .  .  .  CALL INBT(P1(J),P2(J),VAL,INB,FRAC)
  00233          .  .  .  .  IF (INB.EQ.1)
  00234    C     SCANNING LEFT
  00235    30    .  .  .  .  .  X = FLOAT(J-1)*XINC
  00236          .  .  .  .  .  Y = (FLOAT(I-1)+FRAC)*YINC
  00237          .  .  .  .  .  NSCAN = 2
  00238          .  .  .  .  .  ICORN = J*1000 + I
  00239          .  .  .  .  .  IX = X*DSMAX/MAXMAP + 0.5
  00240          .  .  .  .  .  IY = Y*DSMAX/MAXMAP + 0.5
  00241          .  .  .  .  .  CALL VEC(IX,IY)
  00242    D     .  .  .  .  .  WRITE(5,541) NSCAN,IX,IY
  00243          .  .  .  .  .  J = J-1
  00244          .  .  .  .  .  IF(J.LE.1) GO TO 50
  00245          .  .  .  .  .  CALL INBT(P2(J),P2(J+1),VAL,INB,FRAC)
  00246          .  .  .  .  .  IF (INB.EQ.1) GO TO 40
  00247          .  .  .  .  .  CALL INBT(P1(J),P2(J),VAL,INB,FRAC)
  00248          .  .  .  .  .  IF (INB.NE.1) GO TO 20
  00249          .  .  .  .  .  GO TO 30
  00250          .  .  .  .  ...FIN
  00251          .  .  .  .  CALL INBT(P1(J+1),P2(J+1),VAL,INB,FRAC)
  00252          .  .  .  .  IF (INB.EQ.1)
  00253    C     SCANNING RIGHT
  00254    35    .  .  .  .  .  X = FLOAT(J)*XINC
  00255          .  .  .  .  .  Y = (FLOAT(I-1)+FRAC)*YINC
  00256          .  .  .  .  .  NSCAN = 3
  00257          .  .  .  .  .  IX = X*DSMAX/MAXMAP + 0.5
  00258          .  .  .  .  .  IY = Y*DSMAX/MAXMAP + 0.5
  00259          .  .  .  .  .  CALL VEC(IX,IY)
  00260    D     .  .  .  .  .  WRITE(5,541) NSCAN,IX,IY
  00261          .  .  .  .  .  J = J+1
  00262          .  .  .  .  .  IF(J.GE.NODESX) GO TO 50
  00263          .  .  .  .  .  CALL INBT(P2(J),P2(J+1),VAL,INB,FRAC)
  00264          .  .  .  .  .  IF(INB.EQ.1) GO TO 40
  00265          .  .  .  .  .  CALL INBT(P1(J+1),P2(J+1),VAL,INB,FRAC)
  00266          .  .  .  .  .  IF (INB.EQ.1) GO TO 35
  00267          .  .  .  .  .  CALL INBT(P1(J),P1(J+1),VAL,INB,FRAC)
  00268          .  .  .  .  .  IF(INB.NE.1) GO TO 20
  00269    C  HIT BOTTOM LINE
  00270          .  .  .  .  .  NSCAN = 4
  00271          .  .  .  .  .  X = (FLOAT(J-1)+FRAC)*XINC
  00272          .  .  .  .  .  Y = FLOAT(I-1)*YINC
  00273          .  .  .  .  .  ICORN = J*1000+II
  00274          .  .  .  .  .  LAST(NLINE)=ICORN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08140141  PAGE  00006
  00275          .  .  .  .  .  IX = X*DSMAX/MAXMAP + 0.5
  00276          .  .  .  .  .  IY = Y*DSMAX/MAXMAP + 0.5
  00277          .  .  .  .  .  CALL VEC (IX,IY)
  00278    D     .  .  .  .  .  WRITE(5,541) NSCAN,IX,IY
  00279          .  .  .  .  .  GO TO 20
  00280          .  .  .  .  ...FIN
  00281          .  .  .  .  GO TO 20
  00282          .  .  .  ...FIN
  00283    50    .  .  .  CONTINUE
  00284          .  .  ...FIN
  00285          .  ...FIN
  00286          .  CALL ENDSR (ISZ)
  00287          .  IF (ISZ.GT.3500)
  00288          .  .  CALL STRTSB (OVRFLW)
  00289          .  .  ENDDSP = .TRUE.
  00290          .  ...FIN
  00291          ...FIN
  00292    C

         ----------------------------------------

  00293          TO SETUP-DISPLAY-FILES
  00294          .  CALL SUBDF (LABEL,100)
  00295          .  CALL INTENS (7)
  00296          .  CALL CHRTYP (1,0,0)
  00297          .  CALL CHRSCL (3)
  00298          .  CALL APOS (300,1000)
  00299          .  CALL TEXT ('SURFACE CONTOURING')
  00300          .  CALL ENDSR
  00301    C     .
  00302          .  CALL SUBDF (LPLAB,200)
  00303          .  CALL AREA (1)
  00304          .  CALL INTENS (2)
  00305          .  CALL CHRSCL (0)
  00306          .  CALL APOS (3,855)
  00307          .  CALL BOX (155,17)
  00308          .  CALL INTENS (5)
  00309          .  CALL APOS (0,850)
  00310          .  CALL BOX (170,27)
  00311          .  CALL APOS (25,860)
  00312          .  CALL TEXT ('LIGHT PEN OPTIONS:',-3)
  00313          .  CALL MENU (830,40,1,'         RETURN CONTROL',
  00314        1 .  '         SAVE CONTOURS','         CLEAR DISPLAY')
  00315          .  CALL ENDSR
  00316    C     .
  00317          .  CALL SUBDF (MESS1,60,0)
  00318          .  CALL AREA (1)
  00319          .  CALL INTENS (4)
  00320          .  CALL CHRSCL (0)
  00321          .  CALL APOS  (0,100)
  00322          .  CALL TEXT ('ENTER CONTOUR VALUE',-20)
  00323          .  CALL TEXT ('ON DISPLAY KEYBOARD',-20)
  00324          .  CALL TEXT ('(FLOATING POINT)')
  00325          .  CALL AREA (0)
  00326          .  CALL CHRSCL (1)
  00327          .  CALL APOS (0,940)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:40:41  PAGE  00007
  00328           .  CALL TEXT ('CONTOUR VALUE =')
  00329           .  CALL ENDSR
  00330    C      .
  00331           .  CALL SUBDF (OVRFLW,100,1)
  00332           .  CALL CHRSCL (2)
  00333           .  CALL INTENS (7)
  00334           .  CALL BLINK (1)
  00335           .  CALL APOS (250,920)
  00336           .  CALL TEXT ('LINE DISPLAY FILE IS FULL')
  00337           .  CALL BLINK (0)
  00338           .  CALL ENDSR
  00339           ...FIN
  00340           END
```

PROCEDURE CROSS-REFERENCE TABLE

```
00139 DISPLAY-CONTOUR-LINE
      00107

00293 SETUP-DISPLAY-FILES
      00031
```

(FLECS VERSION 22.46)

```
  00341    C
  00342           SUBROUTINE INBT(P1,P2,P,INB,FRAC)
  00343    C
  00344    C  PROGRAM PARAMETERS
  00345    C       P1 & P2 = NODAL VALUES
  00346    C       P       = VALUE TO BE CHECKED
  00347    C       INB   = 0---NOT CONTAINED
  00348    C       INB   = 1--CONTAINED VALUE
  00349    C       FRAC = FRACTIONAL PORTION OF NODE
  00350    C
  00351           INB = 0
  00352           IF (P1.EQ.0..OR.P2.EQ.0.) RETURN
  00353           IF (P1.GT.P2) GO TO 50
  00354           IF (P.LT.P1.OR.P.GE.P2) RETURN
  00355    40     FRAC = (P-P1)/(P2-P1)
  00356           INB = 1
  00357           RETURN
  00358    50     IF (P.GT.P1.OR.P.LE.P2) RETURN
  00359           GO TO 40
  00360           END
```

(FLECS VERSION 22.46)

MXEDIT.FLX

MATRIX EDITOR

```
(FLECS VERSION 22.46)     14-SEP-79   08141114  PAGE  00001

      ----------------------------------------

00001   C ***************  [351,101]MXEDIT.FLX  ***************
00002   C
00003   C
00004   C  PROGRAM ID:  SRV-55-SPA
00005   C
00006   C  PROGRAM DESCRIPTION:  PROGRAM TO DISPLAY X OR Y MATRIX LINE SECTIONS
00007   C                        RATRAN FILE, REPLACE POOR DATA AND UPDATE MATRI
00008   C
00009   C  DATA FILES:
00010   C          NAME                LUN          TYPE          ACCESS
00011   C
00012   C          VS60                 1            --             R/W
00013   C          SURFSPC              4            FLQ            R/W
00014   C          SURFTMP              4            FLQ            R/W
00015   C     VS60                 1
00016   C     SURFSPC     4   FLQ   R/W
00017   C     SURFTMP     4   FLQ   R/W
00018   C
00019   C  LOADING SEQUENCE:
00020   C
00021   C
00022   C  BATTELLE MEMORIAL INSTITUTE
00023   C  PACIFIC NORTHWEST LABORATORIES
00024   C  WATER & LAND RESOURCES DEPT.
00025   C
00026   C  AUTHOR(S):   D.R. FRIEDRICHS
00027   C               V.L. COBURN
00028   C
00029   C  DATE:        INITIAL VERSION   JANUARY 12, 1978
00030   C               CURRENT VERSION   FEBRUARY 2, 1978
00031   C
00032   C
00033   C
00034         BYTE FNAME,SRCMAP,DSPMAP
00035         REAL MAXMAP,MINY,MAXY
00036   C
00037         INTEGER OUT,BOXV
00038         LOGICAL EXPND,VALSW,AUTO
00039         DIMENSION ZZ(5,5),MEM(1),IND(9),NDIGIT(78)
00040         DIMENSION SVAL(256)
00041   C
00042         COMMON/CMXE/ MINDX,MAXDX,MINDY,MAXDY
00043         COMMON/DSP1/ MAIN(100),LINE1(400),LINE2(400),LINE3(400),BOXV(20),
00044        1   OUT(300),LSCAN(60),LSCAN1(60),LSCAN2(60),LSCAN3(60),LSCAN4(75),
00045        2   MESS1(25),MESS2(25),MESS3(25),MESS4(25),MESS5(25),MESS6(25),
00046        3   MESS7(25),MESS8(25),MESS9(25)
00047         COMMON/DSP2/ IGRAPH(100),NGRAPH(1200),LABL1(20),LABL2(20),
00048        1   LBUT(200),IVAR(100)
00049         COMMON/TND/ INITDZ,DZXMAX,DZYMAX,DZMAX,NODESX,NODESY,XINC,YINC,
00050        1            NSX,TOTSEC,TOTX,FNAME(30),SRCMAP(30),DSPMAP(30),
00051        2            XMAX,YMAX,MAXMAP,DSMAX,IARG,WRKMIN,WRKMAX,LINE
00052   C
00053   C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:41:14  PAGE  00002
  00054          EQUIVALENCE (INITDZ,NDIGIT(1))
  00055    C
  00056    C
  00057    C
  00058          IO1 = 0
  00059          IV1 = 0
  00060          IL1 = 0
  00061          IL2 = 0
  00062          IL3 = 0
  00063          IS1 = 1
  00064    C
  00065          CALL MAINDF (MAIN,100,0,1)
  00066          CALL CLKOFF
  00067    C
  00068          CALL MXDSP
  00069          CALL OPFILE (4,'SURFSPC',1,,IPNT0)
  00070          CALL OPR (IPNT0,0,,NDIGIT,78,)
  00071          CALL OPFILE (4,'SURFTMP',TOTSEC,IPNT1)
  00072          NARG=IARG
  00073          IARG=12
  00074    C
  00075          NARG=11
  00076          WRKMIN=0.
  00077          WRKMAX=0.
  00078    C
  00079          NSY=(NODESY+127)/128
  00080    C     CALL MXSWAP(NODESX,NODESY,DSKX,DSKY,KDA1,KDA2,NDU)
  00081    C ***  SEE IF MIN & MAX MATRIX VALUEW HAVE BEEN CALCULATED
  00082          CALL STRTSB (IGRAPH,LABL1,LSCAN)
  00083          IF (WRKMIN.EQ.0.. AND .WRKMAX.EQ.0.) CALCULATE-MIN-AND-MAX-VALUES
  00084          PAVE=(WRKMAX+WRKMIN)/2.
  00085    C  CHECK FOR RETURN FROM CONTOUR
  00086          WHEN (NARG.EQ.13)
  00087          .  IF(MODE.EQ.1) LINEY=LINE
  00088          .  IF(MODE.EQ.2) LINEX=LINE
  00089          ...FIN
  00090          ELSE
  00091          .  MODE = 1
  00092          .  LINEY= 1
  00093          .  LINEX= 0
  00094          .  MINX = 1
  00095          .  MAXX = NODESX
  00096          .  MINY = WRKMIN
  00097          .  MAXY = WRKMAX
  00098          .  VALSW=.FALSE.
  00099          .  EXPND=.FALSE.
  00100          .  AUTO =.FALSE.
  00101          .  LINC = 1
  00102          ...FIN
  00103          DISPLAY-VARIABLES
  00104          CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
  00105    C
  00106    C************DISPLAY INTERRUPTS************
  00107    C
  00108    C
  00109    C
```

```
(FLECS VERSION 22.46)     14-SEP-79   08141814  PAGE  00003
  00110   19    CALL GRATTN (0,ITYP,'LP')
  00111         IF(AUTO) AUTO-SCAN-AND-DISPLAY-LINE
  00112         IF(ITYP.NE.'LP') GO TO 19
  00113   C *** LIGHT PEN INTERRUPTS ***
  00114         CALL LTPEN (ID,ITIP)
  00115         IF(ITIP.EQ.1)
  00116         .  CONDITIONAL
  00117   C *** HORIZONTAL LINE ***
  00118         .  .  (ID.EQ.1)
  00119         .  .  .  MODE=1
  00120         .  .  .  CALL STRTSB (MESS1)
  00121         .  .  .  READ(2,221) LINEY
  00122   221   .  .  .  FORMAT(I4)
  00123         .  .  .  CALL STOPSB (MESS1)
  00124         .  .  .  LINE=LINEY
  00125         .  .  .  LINEX=0
  00126         .  .  .  DISPLAY-VARIABLES
  00127         .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00128         .  .  ...FIN
  00129   C *** VERTICAL LINE ***
  00130         .  .  (ID.EQ.2)
  00131         .  .  .  MODE=2
  00132         .  .  .  CALL STRTSB (MESS2)
  00133         .  .  .  READ(2,221) LINEX
  00134         .  .  .  CALL STOPSB (MESS2)
  00135         .  .  .  LINE=LINEX
  00136         .  .  .  LINEY=0
  00137         .  .  .  DISPLAY-VARIABLES
  00138         .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00139         .  .  ...FIN
  00140   C *** NODE NUMBER ***
  00141         .  .  (ID.EQ.3)
  00142         .  .  .  CALL STRTSB (MESS3)
  00143         .  .  .  READ(2,221) NODE
  00144         .  .  .  CALL STOPSB (MESS3)
  00145         .  .  .  DISPLAY-VARIABLES
  00146         .  .  ...FIN
  00147   C *** CHANGE VALUE ***
  00148         .  .  (ID.EQ.4)
  00149         .  .  .  CALL STRTSB (MESS4)
  00150         .  .  .  READ(2,222) VALUE
  00151   222   .  .  .  FORMAT(F12.0)
  00152         .  .  .  CALL STOPSB (MESS4)
  00153         .  .  .  CHANGE-VALUE-AT-NODE
  00154         .  .  .  DISPLAY-VARIABLES
  00155         .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00156         .  .  ...FIN
  00157   C *** DISPLAY VALUE AND SURROUNDING VALUES ***
  00158         .  .  (ID.EQ.5)
  00159         .  .  .  VALSW=.TRUE.
  00160         .  .  .  DISPLAY-VALUE-AND-SURROUNDING-VALUES
  00161         .  .  ...FIN
  00162   C *** DISPLAY GRAPH ***
  00163         .  .  (ID.EQ.6)
  00164         .  .  .  VALSW=.FALSE.
  00165         .  .  .  CALL STOPSB (OUT,BOXV)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08141114  PAGE  00004
  00166          .  .  .  CALL STRTSB (IGRAPH,NGRAPH,LINE1,LINE2,LINE3,LSCAN)
  00167          .  .  ...FIN
  00168    C     .  .
  00169    C     .  .
  00170    C *** EXPAND Y OR X SCALE
  00171          .  .  (ID.EQ.7.OR.ID.EQ.8)
  00172          .  .  .  EXPAND=.TRUE.
  00173          .  .  .  SELECT (ID)
  00174          .  .  .  .  (7)
  00175          .  .  .  .  .  CALL STRTSB (MESS5)
  00176          .  .  .  .  .  READ(2,222) MINY
  00177          .  .  .  .  .  CALL STOPSB (MESS5)
  00178          .  .  .  .  .  CALL STRTSB (MESS6)
  00179          .  .  .  .  .  READ(2,222) MAXY
  00180          .  .  .  .  .  CALL STOPSB (MESS6)
  00181          .  .  .  .  ...FIN
  00182          .  .  .  .  (8)
  00183          .  .  .  .  .  CALL STRTSB (MESS7)
  00184          .  .  .  .  .  READ(2,221) MINX
  00185          .  .  .  .  .  CALL STOPSB (MESS7)
  00186          .  .  .  .  .  CALL STRTSB (MESS8)
  00187          .  .  .  .  .  READ(2,221) MAXX
  00188          .  .  .  .  .  CALL STOPSB (MESS8)
  00189          .  .  .  .  ...FIN
  00190          .  .  .  ...FIN
  00191          .  .  .  CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
  00192          .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00193          .  .  ...FIN
  00194    C *** NORMAL SCALE ***
  00195          .  .  (ID.EQ.9)
  00196          .  .  .  MINY = WRKMIN
  00197          .  .  .  MAXY = WRKMAX
  00198          .  .  .  MINX = 1
  00199          .  .  .  IF (MODE.EQ.1) MAXX = NODESX
  00200          .  .  .  IF (MODE.EQ.2) MAXX = NODESY
  00201          .  .  .  CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
  00202          .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00203          .  .  .  EXPND=.FALSE.
  00204          .  .  ...FIN
  00205    C *** GO TO CONTOURING PROGRAM OR RETURN TO CONTROL ***
  00206          .  .  (ID.EQ.10.OR.ID.EQ.11)
  00207          .  .  .  CALL DPW (IPNT0,0,NDIGIT,78,)
  00208          .  .  .  WHEN (ID.EQ.10) TASK=RAD50('CONTUR')
  00209          .  .  .  ELSE
  00210          .  .  .  .  TASK=RAD50('APPLIC')
  00211          .  .  .  ...FIN
  00212          .  .  .  CALL REQUES (TASK,,IDS)
  00213          .  .  .  CALL EXIT
  00214          .  .  ...FIN
  00215    C *** ENTER AUTO SCAN ***
  00216          .  .  (ID.EQ.12)
  00217          .  .  .  CALL STOPSB (LINE1,LINE2,LINE3,LSCAN)
  00218          .  .  .  CALL WAIT (2,2,MMM)
  00219          .  .  .  CALL STRTSB (LSCAN1)
  00220          .  .  .  LINC = 1
  00221          .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:41:14  PAGE  00005
  00222   C *** SELECT HORIZONTAL OR VERTICAL SCAN ***
  00223           .  .  (ID.EQ.13. OR .ID.EQ.14)
  00224           .  .  .  MINY = WRKMIN
  00225           .  .  .  MAXY = WRKMAX
  00226           .  .  .  MINX = 1
  00227           .  .  .  SELECT (ID)
  00228           .  .  .  .  (13)
  00229           .  .  .  .  .  MODE = 1
  00230           .  .  .  .  .  LINEY = 1
  00231           .  .  .  .  .  MAXX = NODESX
  00232           .  .  .  .  ...FIN
  00233           .  .  .  .  (14)
  00234           .  .  .  .  .  MODE = 2
  00235           .  .  .  .  .  LINEX = 1
  00236           .  .  .  .  .  MAXX = NODESY
  00237           .  .  .  .  ...FIN
  00238           .  .  .  ...FIN
  00239           .  .  .  CALL STOPSB (LSCAN1)
  00240           .  .  .  CALL STRTSB (LSCAN2)
  00241           .  .  .  DELAY=.FALSE.
  00242           .  .  .  READ(2,230) ANS
  00243   230     .  .  .  FORMAT(A1)
  00244           .  .  .  CALL STOPSB (LSCAN2)
  00245           .  .  .  IDLAY = 500
  00246           .  .  .  IF(ANS.EQ.'Y')
  00247           .  .  .  .  CALL STRTSB (LSCAN3)
  00248           .  .  .  .  READ(2,231) DLAY
  00249   231     .  .  .  .  FORMAT(F5.0)
  00250           .  .  .  .  IDLAY = DLAY*1000.
  00251           .  .  .  .  DELAY=.TRUE.
  00252           .  .  .  .  CALL STOPSB (LSCAN3)
  00253           .  .  .  ...FIN
  00254           .  .  .  CALL STRTSB (LSCAN4)
  00255           .  .  .  CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
  00256           .  .  ...FIN
  00257   C *** BEGIN SCAN ***
  00258           .  .  (ID.EQ.15)
  00259           .  .  .  AUTO = .TRUE.
  00260           .  .  ...FIN
  00261   C *** REVERSE SCAN ***
  00262           .  .  (ID.EQ.16)
  00263           .  .  .  AUTO = .TRUE.
  00264           .  .  .  LINC = LINC*(-1)
  00265           .  .  ...FIN
  00266   C *** STOP SCAN ***
  00267           .  .  (ID.EQ.17)
  00268           .  .  .  AUTO=.FALSE.
  00269           .  .  ...FIN
  00270   C *** LEAVE AUTO SCAN ***
  00271           .  .  (ID.EQ.18)
  00272           .  .  .  AUTO = .FALSE.
  00273           .  .  .  CALL STOPSB (LSCAN4)
  00274           .  .  .  CALL STRTSB (LSCAN)
  00275           .  .  .  CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00276           .  .  ...FIN
  00277           .  ...FIN
```

```
(FLECS VERSION 22.06)     14-SEP-79   08101114  PAGE  00006
  00278          ...FIN
  00279    C
  00280          GO TO 19

         ----------------------------------------

  00281          TO CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
  00282    C   ********CALCULATE GRAPH RESCALING IF NECCESSARY*****
  00283    50      .   SELECT (MODE)
  00284            .   .   (1)
  00285            .   .   .   LINE=LINEY
  00286            .   .   .   KOA=IPNT1
  00287            .   .   .   NS=NSX
  00288            .   .   .   MAXLIN=NODESY
  00289            .   .   .   NPTS=NODESX
  00290            .   .   .   TOTWRD = NPTS*2
  00291            .   .   .   CALL STOPSB (LABL2)
  00292            .   .   .   CALL STRTSB (LABL1)
  00293            .   .   ...FIN
  00294            .   .   (2)
  00295            .   .   .   LINE=LINEX
  00296            .   .   .   KOA=IPNT2
  00297            .   .   .   NS=NSY
  00298            .   .   .   MAXLIN=NODESX
  00299            .   .   .   NPTS=NODESY
  00300            .   .   .   TOTWRD = NPTS*2
  00301            .   .   .   CALL STOPSB (LABL1)
  00302            .   .   .   CALL STRTSB (LABL2)
  00303            .   .   ...FIN
  00304            .   ...FIN
  00305    C  SEE IF IN EXPAND MODE
  00306            .   INCR=0
  00307            .   SINC=(MAXY-MINY)/10.
  00308            .   IF(SINC.GE.1.) INCR=SINC
  00309    C       .
  00310    C ****SETUP NEW SCALE FOR GRAPH*****
  00311    51      .   CALL SUBOF (NGRAPH,1200,I51)
  00312            .   I51 = 3
  00313            .   CALL CHRSCL (0)
  00314            .   XSCALE=FLOAT(MAXDX-MINDX)/FLOAT(MAXX-MINX)
  00315            .   II=MINX-(MINX/10)*10-1
  00316            .   DO (I=MINX,MAXX)
  00317            .   .   II=II+1
  00318            .   .   IX=MINDX+FLOAT(I-MINX)*XSCALE
  00319            .   .   IF (II.EQ.10)
  00320            .   .   .   CALL INTENS (7)
  00321            .   .   .   CALL APOS (IX-10,MINDY-15)
  00322            .   .   .   CALL INMOR (I,'(I3)')
  00323            .   .   .   II=0
  00324            .   .   ...FIN
  00325            .   .   WHEN (II.EQ.0) CALL INTENS (6)
  00326            .   .   ELSE
  00327            .   .   .   CALL INTENS (3)
  00328            .   .   ...FIN
  00329            .   .   CALL APOS (IX,MINDY)
  00330            .   .   CALL VEC (IX,MAXDY)
```

```
(FLECS VERSION 22.46)     14-SEP-79   08141114  PAGE  00007
  00331          .  ...FIN
  00332    C     CALCULATE Y LINES
  00333          .  YSCALE=FLOAT(MAXDY-MINDY)/(MAXY-MINY)
  00334          .  IF(INCR.EQ.0) GO TO 52
  00335          .  IF(MAXY.GT.30000.) GO TO 52
  00336          .  IX=20
  00337          .  MY1=MINY+.001
  00338          .  MY2=MAXY+.001
  00339          .  DO (J=MY1,MY2,INCR)
  00340          .  .  IY=MINDY+FLOAT(J-MY1)*YSCALE
  00341          .  .  CALL INTENS (7)
  00342          .  .  CALL CHRSCL (0)
  00343          .  .  CALL APOS (IX,IY-5)
  00344          .  .  CALL INMBR (J,'(I10)')
  00345          .  .  CALL APOS (MINDX,IY)
  00346          .  .  CALL VEC  (MAXDX,IY)
  00347          .  ...FIN
  00348          .  GO TO 5260
  00349    C  VERY SMALL OR VERY LARGE Y SCALE
  00350    52    .  IX=0
  00351          .  DIFFY=ABS(MAXY-MINY)
  00352          .  DO (J=1,11)
  00353          .  .  YY=SINC*(J-1)
  00354          .  .  IY=MINDY+YY*YSCALE
  00355          .  .  CALL INTENS (7)
  00356          .  .  CALL CHRSCL (0)
  00357          .  .  CALL APOS (IX,IY-5)
  00358          .  .  YYY=MINY+YY
  00359          .  .  CONDITIONAL
  00360          .  .  .  (DIFFY.GE.100000.) CALL FNMBR (YYY,'(F10.0)')
  00361          .  .  .  (DIFFY.GE.10000.) CALL FNMBR (YYY,'(F10.2)')
  00362          .  .  .  (DIFFY.GE.100.) CALL FNMBR (YYY,'(F10.4)')
  00363          .  .  .  (OTHERWISE) CALL FNMBR (YYY,'(F10.5)')
  00364          .  .  ...FIN
  00365          .  .  CALL APOS (MINDX,IY)
  00366          .  .  CALL VEC (MAXDX,IY)
  00367          .  ...FIN
  00368    5260  .  CALL ENDSB
  00369          .  IF(.NOT.VALSW) CALL STRTSB (NGRAPH)
  00370          ...FIN

         ----------------------------------------

  00371          TO CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
  00372    53    .  IF(MODE.EQ.1) NTOP=(NODESY-1)*NS
  00373          .  IF(MODE.EQ.2) NTOP=(NODESX-1)*NS
  00374          .  IND(3)=(LINE-1)*NS
  00375          .  IND(2)=IND(3)-NS
  00376          .  IND(4)=IND(3)+NS
  00377          .  DO (II=2,4)
  00378          .  .  ADR=IND(II)
  00379          .  .  IF (IND(II).GE.0. AND .IND(II).LE.NTOP)
  00380          .  .  .  CALL DPR (KDA,ADR,SVAL,TOTWRD)
  00381          .  .  ...FIN
  00382          .  .  SELECT (II)
  00383          .  .  .  (2)
```

```
(FLECS VERSION 22.461)      14-SEP-79   08:41:14  PAGE  00008
  00384          .  .  .  .  CALL SUBDF (LINE1,400,IL1)
  00385          .  .  .  .  IL1=3
  00386          .  .  .  .  CALL INTENS (7)
  00387          .  .  .  .  CALL LINTYP (2)
  00388          .  .  .  ...FIN
  00389          .  .  .  (3)
  00390          .  .  .  .  CALL SUBDF (LINE2,400,IL2)
  00391          .  .  .  .  IL2=3
  00392          .  .  .  .  CALL INTENS (7)
  00393          .  .  .  .  CALL LINTYP (0)
  00394          .  .  .  ...FIN
  00395          .  .  .  (4)
  00396          .  .  .  .  CALL SUBDF (LINE3,400,IL3)
  00397          .  .  .  .  IL3=3
  00398          .  .  .  .  CALL INTENS (7)
  00399          .  .  .  .  CALL LINTYP (1)
  00400          .  .  .  ...FIN
  00401          .  .  ...FIN
  00402          .  .  IF(IND(II).GE.0. AND .IND(II).LE.NTOP)
  00403          .  .  .  DO (I=MINX,MAXX)
  00404          .  .  .  .  IX=MINDX+FLOAT(I-MINX)*XSCALE
  00405          .  .  .  .  WHEN (SVAL(I).EQ.0.) Z=MINY
  00406          .  .  .  .  ELSE
  00407          .  .  .  .  .  Z = SVAL(I)
  00408          .  .  .  .  .  IF(Z.GT.MAXY) Z=MAXY
  00409          .  .  .  .  .  IF(Z.LT.MINY) Z=MINY
  00410          .  .  .  .  ...FIN
  00411          .  .  .  .  IY=MINDY+(Z-MINY)*YSCALE
  00412          .  .  .  .  IF(I.EQ.MINX) CALL APOS (IX,IY)
  00413          .  .  .  .  IF(I.NE.MINX) CALL VEC (IX,IY)
  00414          .  .  .  ...FIN
  00415          .  .  ...FIN
  00416          .  .  CALL ENDSB
  00417          .  .  IF(.NOT.VALSW)
  00418          .  .  .  SELECT (II)
  00419          .  .  .  .  (2) CALL STRTSB (LINE1)
  00420          .  .  .  .  (3) CALL STRTSB (LINE2)
  00421          .  .  .  .  (4) CALL STRTSB (LINE3)
  00422          .  .  .  ...FIN
  00423          .  .  ...FIN
  00424          .  ...FIN
  00425          .  IF(VALSW) DISPLAY-VALUE-AND-SURROUNDING-VALUES
  00426          ...FIN
  00427    C
  00428    C  OUTPUT VALUES AT GIVEN NODE NO.

         ----------------------------------------

  00429          TO DISPLAY-VALUE-AND-SURROUNDING-VALUES
  00430          .  IND(1)=IND(2)-NS
  00431          .  IND(5)=IND(4)+NS
  00432          .  DO (J=1,5)
  00433          .  .  IF(IND(J).GE.0. AND .IND(J).LE.NTOP)
  00434          .  .  .  ADR=IND(J)
  00435          .  .  .  CALL DPR (KOA,ADR,SVAL,TOTWRD)
  00436          .  .  ...FIN
```

```
(FLECS VERSION 22.46)     14-SEP-79   08:41:14  PAGE  00009
  00437          .  .  NB=NODE-2
  00438          .  .  NT=NODE+2
  00439          .  .  II=0
  00440          .  .  DO (I=NB,NT)
  00441          .  .  .  II=II+1
  00442          .  .  .  WHEN (IND(J).LT.0. OR .IND(J).GT.NTOP) ZZ(J,II)=0.0
  00443          .  .  .  ELSE
  00444          .  .  .  .  ZZ(J,II)=SVAL(I)
  00445          .  .  .  ...FIN
  00446          .  .  ...FIN
  00447          .  ...FIN
  00448          .  CALL STOPSB (LINE1,LINE2,LINE3,IGRAPH,NGRAPH)
  00449          .  CALL STOPSB (LSCAN,LSCAN1,LSCAN2,LSCAN3,LSCAN4)
  00450          .  CALL STRTSB (BOXV)
  00451          .  CALL SUBDF (OUT,300,101)
  00452          .  I01=2
  00453          .  IY = 640
  00454          .  DO (J=5,1,-1)
  00455          .  .  CALL APDS (0,IY)
  00456          .  .  DO (I=1,5)
  00457          .  .  .  ZTMP = ZZ(J,I)
  00458          .  .  .  CALL FNMBR (ZTMP,'(1PE14.5)')
  00459          .  .  ...FIN
  00460          .  .  IY = IY-58
  00461          .  ...FIN
  00462          .  CALL ENDSB
  00463          ...FIN

         ------------------------------------

  00464          TO CHANGE-VALUE-AT-NODE
  00465    65    .  SELECT (MODE)
  00466          .  .  (1)
  00467          .  .  .  III=LINEY
  00468          .  .  .  II2=NODE
  00469          .  .  ...FIN
  00470          .  .  (2)
  00471          .  .  .  III=NODE
  00472          .  .  .  II2=LINEX
  00473          .  .  ...FIN
  00474          .  ...FIN
  00475    C     .
  00476          .  ADR=(III-1)*NSX
  00477          .  CALL DPR (IPNT1,ADR,SVAL,TOTX)
  00478          .  SVAL(II2)=VALUE
  00479          .  CALL DPW (IPNT1,ADR,SVAL,TOTX)
  00480    C     .
  00481    C     .  ADR=(II2-1)*NSY
  00482    C     .  CALL DPR (IPNT2,ADR,SVAL,TOTY)
  00483    C     .  SVAL(III)=VALUE
  00484    C     .  CALL DPW (IPNT2,ADR,SVAL,TOTY)
  00485          ...FIN

         ------------------------------------

  00486          TO CALCULATE-MIN-AND-MAX-VALUES
```

```
(FLECS VERSION 22.46)     14-SEP-79   08141114  PAGE  00010
  00487          .  WRKMIN = 100000.
  00488          .  WRKMAX = -100000.
  00489          .  DO (J=1,NODESY)
  00490          .  .  ADR = J-1
  00491          .  .  CALL DPR (IPNT1,ADR,SVAL,TOTX)
  00492          .  .  DO (K=1,NODESX)
  00493          .  .  .  IF (SVAL(K).NE.0.)
  00494          .  .  .  .  IF (SVAL(K).LT.WRKMIN) WRKMIN=SVAL(K)
  00495          .  .  .  .  IF (SVAL(K).GT.WRKMAX) WRKMAX=SVAL(K)
  00496          .  .  .  ...FIN
  00497          .  .  ...FIN
  00498          .  ...FIN
  00499          ...FIN

         ----------------------------------------

  00500          TO DISPLAY-VARIABLES
  00501          .  CALL SUBDF (IVAR,100,IV1)
  00502          .  IV1 = 2
  00503          .  CALL AREA (1)
  00504          .  CALL INTENS (3)
  00505          .  CALL CHRSCL (0)
  00506          .  CALL APOS (163,820)
  00507          .  CALL INMBR (LINEY,'(I3)',-40)
  00508          .  CALL INMBR (LINEX,'(I3)',-40)
  00509          .  CALL INMBR (NODE,'(I3)')
  00510          .  CALL APOS (114,700)
  00511          .  CALL FNMBR (VALUE,'(1PE12.4)')
  00512          .  CALL AREA (0)
  00513          .  CALL ENDSR
  00514          ...FIN

         ----------------------------------------

  00515          TO AUTO-SCAN-AND-DISPLAY-LINE
  00516          .  IF(LINE.LT.1) LINE = 1
  00517          .  IF(LINE.GT.MAXLIN) LINE = MAXLIN
  00518          .  ADR = (LINE-1)*NS
  00519          .  CALL DPR(KDA,ADR,SVAL,TOTWRD)
  00520          .  CALL WAIT (IDLAY,1,MMM)
  00521          .  SELECT (MODE)
  00522          .  .  (1)
  00523          .  .  .  LINEY = LINE
  00524          .  .  .  LINEX = 0
  00525          .  .  ...FIN
  00526          .  .  (2)
  00527          .  .  .  LINEX = LINE
  00528          .  .  .  LINEY = 0
  00529          .  .  ...FIN
  00530          .  ...FIN
  00531          .  CALL SUBDF (LINE1,400,IL1)
  00532          .  IL1 = 3
  00533          .  CALL INTENS (7)
  00534          .  CALL LINTYP (2)
  00535          .  DO (I=1,MAXX)
  00536          .  .  IX=MINDX+FLOAT(I-MINX)*XSCALE
```

```
(FLECS VERSION 22.46)     14-SEP-79   08141114  PAGE  00011
  00537          .  .  WHEN (SVAL(I).EQ.0.) Z=MINY
  00538          .  .  ELSE
  00539          .  .  .  Z = SVAL(I)
  00540          .  .  .  IF(Z.GT.MAXY) Z=MAXY
  00541          .  .  .  IF(Z.LT.MINY) Z=MINY
  00542          .  .  ...FIN
  00543          .  .  IY=MINDY+(Z-MINY)*YSCALE
  00544          .  .  IF(I.EQ.MINX) CALL APOS (IX,IY)
  00545          .  .  IF(I.NE.MINX) CALL VEC (IX,IY)
  00546          .  ...FIN
  00547          .  CALL ENDSB
  00548          .  CALL STRTSB (LINE1)
  00549          .  DISPLAY-VARIABLES
  00550          .  LINE = LINE+LINC
  00551          .  IF(LINE.LT.1)
  00552          .  .  LINE = 2
  00553          .  .  LINC = +1
  00554          .  ...FIN
  00555          .  IF(LINE.GT.MAXLIN)
  00556          .  .  LINC = MAXLIN-1
  00557          .  .  LINC = -1
  00558          .  ...FIN
  00559          ...FIN
  00560          END

         ----------------------------------------

               PROCEDURE CROSS-REFERENCE TABLE

         00515 AUTO-SCAN-AND-DISPLAY-LINE
               00111

         00371 CALCULATE-MATRIX-LINES-FOR-GRAPH-DISPLAY
               00127  00138  00155  00192  00202  00275

         00486 CALCULATE-MIN-AND-MAX-VALUES
               00083

         00281 CALCULATE-PARAMETERS-AND-DISPLAY-BACKGROUND-GRAPH
               00104  00191  00201  00255

         00464 CHANGE-VALUE-AT-NODE
               00153

         00429 DISPLAY-VALUE-AND-SURROUNDING-VALUES
               00160  00425

         00500 DISPLAY-VARIABLES
               00103  00126  00137  00145  00154  00549


         (FLECS VERSION 22.46)

         ----------------------------------------
```

PNL-3161-2
UC-70

DISTRIBUTION

No. of Copies

OFFSITE

A. A. Churm
DOE Chicago Patent Group
9800 South Cass Avenue
Argonne, IL 60439

27 DOE Technical Information Center

2 Argonne National Laboratory
Reference Library
9800 South Cass Avenue
Argonne, IL 60439

G. E. Barr
Div. 1141
Sandia Laboratories
Albuquerque, NM 87115

Battelle Memorial Institute
Office of Nuclear Waste Isolation
Attn: Beverly Rawles
505 King Avenue
Columbus, OH 43201

John Bird
Geology Department
Cornell University
Ithaca, NY 14853

J. D. Bredehoeft
U.S. Geological Survey
Water Resources Division
Reston, VA 22092

2 Brookhaven National Laboratory
Reference Section
Information Division
Upton, Long Island, NY 11973

No. of Copies

J. E. Campbell
Div. 54.3
Sandia Laboratories
Albuquerque, NM 87115

20 Wayne A. Carbiener
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

H. C. Claiborne
Union Carbide Corp.
P.O. Box Y
Oak Ridge, TN 39380

Neville G. W. Cook
Dept. of Materials Science and Mineral Engineering
Hearst Mining Building
University of California
Berkeley, CA 94720

Carl R. Cooley
DOE Office of Waste Management
Washington, D. C. 20545

Jared Davis
Nuclear Regulatory Commission
Washington, DC 20555

George DeBuchananne
U. S. Geological Survey
Reston, VA 22092

R. T. Dillon
Div. 5413
Sandia Laboratories
Albuquerque, NM 87115

No. of Copies

J. O. Duguid
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Ave.
Columbus, OH 43201

Environmental Protection Agency
Office of Radiation Programs
Technical Assessment Division AW559
Washington, DC 20460

Geraghty and Miller, Inc.
44 Sintsink Dr. East
Port Washington, NY 11050

Robert M. Garrells
Dept. of Geologic Sciences
Northwestern University
Princeton, NY 08540

W. G. Gray
Dept. of Civil Engineering
Princeton University
Princeton, NJ 08540

D. B. Grove
U.S. Geological Survey
WRD
Mail Stop 413
Denver Federal Center
Denver, CO 80225

John Handin, Director
Center for Tectomophysics
Texas A>M University
College Station, TX 77840

Colin A. Heath
DOE Division of Waste Management
Washington, DC 20545

William M. Hewitt
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

No. of Copies

Peter L. Hofmann
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

Muzaffer Kehnemuyi
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

John F. Kircher
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

Ronald B. Lantz
Inter-Environmental Consultants
11511 Katy Freeway
Houston, TX 77079

2 Lawrence Berkeley Laboratory
Reference Library
University of California
Berkeley, CA 94720

2 Lawrence Livermore Laboratory
Reference Library
P. O. Box 808
Livermore, CA 94550

D. I. Leap
U.S. Geological Survey
Water Resources Division
MS 416
Denver Federal Center
Denver, CO 80225

S. E. Logan
Los Alamos Technical Associates, Inc.
P.O. Box 410
Los Alamos, NM 87544

No. of Copies

2 Los Alamos Scientific Laboratory
Reference Library
P.O. Box 1663
Los Alamos, NM 87544

John Lyons
Dept. of Earth Sciences
Dartmouth College
Hanover, NH 03755

R. D. MacNish
U.S. Geological Survey
Water Resources Div.
301 W. Congress
Tucson, AR 85701

J. B. Martin
Asst. Director for Radioactive Waste Mgmt Branch
NRC Division of Materials and Fuel Cycle Facility Licensing
Washington, DC 20555

John T. McGinnis
Office of Nuclear Waste Isolation
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201

Sheldon Meyers
DOE Office of Nuclear Waste Management
Washington, DC 20545

P. A. Mote
Bechtel National, Inc.
50 Beale St.
San Francisco, CA 94119

2 Barry Naft
NUS Corporation
4 Research Place
Rockville, MD 20805

T. N. Narasimhan
Lawrence Berkeley Laboratory
Berkeley, CA 94720

No. of Copies

J. O. Neff
Department of Energy
Columbus Program Office
505 King Avenue
Columbus, OH 43201

2 Neil A. Norman
Environmental Sciences Department
Bechtel National Inc.
P.O. Box 3965
San Francisco, CA 94105

2 Oak Ridge National Laboratory
Central Research Library
Document Reference Section
Oak Ridge, TN 37830

Frank L. Parker
Dept. of Environmental Eng.
Vanderbilt University
Nashville, TN 37235

L. W. Picking
Stone and Webster Eng. Corp.
245 Summer St.
Boston, MA 02107

George Pinder
Dept. of Civil Engineering
Princeton University
Princeton, NJ 08540

Gary Robbins
U.S. Nuclear Regulatory Commission
Division of Waste Management
Washington, DC 20555

2 Savannah River Laboratory
Reference Library
Aiken, SC 29801

G. Segol
Bechtel National, Inc.
50 Beale St.
San Francisco, CA 94119

No. of
Copies

Raymond Siever
Dept. of Geological Sciences
Harvard University
Cambridge, MA 02138

Howard P. Stephens
Sandia Laboratories
P.O. Box 5800
Albuquerque, NM 87115

David B. Stewart
U.S. Geological Survey
National Center 959
Reston, VA 22092

Lewis D. Thorson
Lawrence Livermore Laboratory
Box 803, L-224
Livermore, CA 94550

N. P. Timofeef
State University of New York
Binghampton, NY 13901

Wendal Weart
Division 1140
Sandia Laboratories
Albuquerque, NM 87115

Robert Williams
Electric Power Research Inst.
3412 Hillview Avenue
P.O. Box 10412
Palo Alto, CA 94304

Paul A. Witherspoon
Lawrence Berkeley Laboratory
University of California
Berkeley, CA 94710

FOREIGN

D'Allessandro Avogadro
Commission of European
Communities
Joint Research Centre
I-21020 Ispra (Varese)
ITALY

No. of
Copies

V. K. Barwell
Environmental Research Branch
Atomic Energy of Canada, Ltd.
Chalk River, Ontario, KOJIJO
CANADA

Center for Atomic Energy
Documentation (2AED)
Attn: Dr. Bell
Postfach 3640
D-7500 Karlsruhe
F.R. of GERMANY

J. A. Cherry
Dept. of Earth Sciences
Waterloo, Ontario
CANADA

Ferruccio Gera
Radiation Protection and
Waste Management Division
Nuclear Energy Agency/OECD
38 boulevard Suchet
75016 Paris
FRANCE

K. H. Hubenthal
Bundesministerium fur Forschung
und Technologie
Stressemannstrasse 2
Postfach 200706
D-5300 Bonn
F.R. of GERMANY

2 INIS Clearinghouse
International Atomic Energy
Agency
P.O. Box 590
A-1011, Vienna
AUSTRIA

Klaus Kuhn
Institute fur Tiefagerung
Wissenschaftliche Abteilung
Berliner Strasse 2
3392 Clausthal - Zellerfeld
F.R. of GERMANY

No. of
Copies

Hans W. Levi
Hahn-Meitner-Institut fur
Kernforschung
Glienicker Strasse 100
1000 Berlin 39
F.R. of GERMANY

Library
Studsvik Energiteknik AB
S-611 01 Nykoping
SWEDEN

Franz Peter Oesterle
Physikalisch-Chemische
Bundesanstalt
Bundesallee 100
D-3300 Braunschweig
F.R. of GERMANY

Tonis Papp
Karnbranslesakerhet
Fack. 10240
Stockholm
SWEDEN

Egbert Schapermeier
Battelle-Institute e.V.
Am Romerhof 35
D-6000 Frankfurt am Main 90
F.R. of GERMANY

Tjalle Vandergraff
Atomic Energy of Canada Limited
Whiteshell Nuclear Research
Establishment
Pinawa, Manitoba ROE 1LQ
CANADA

ONSITE

4 DOE Richland Operations Office

O. J. Elgert
H. E. Ransom
J. J. Schreiber
F. R. Standerfer

No. of
Copies

3 Rockwell Hanford Operations

R. C. Arnett
R. A. Deju
Rockwell Document Control

61 Pacific Northwest Laboratory

G. L. Benson
D. L. Bradley
A. Brandstetter (10)
R. W. Bond
J. S. Burlison
D. B. Cearlock
M. O. Cloninger
C. R. Cole
F. H. Dove
D. W. Dragnich
D. R. Friedrichs
S. R. Gupta
M. A. Harwell
B. W. Howes
J. H. Jarrett
F. E. Kaszeta
M. R. Kreiter
J. S. Light
R. W. Nelson
A. M. Platt
J. R. Raymond (10)
A. E. Reisenauer
R. J. Serne
C. S. Simmons
J. K. Soldat
J. F. Washburn
Technical Information Library
(5)
Publishing Coordination Ro (2)
Water and Land Resources
Department Library (10)

MODEL USERS FORM

If you wish to be on the distribution list for future model computer code and report revisions, please fill out the form below and return it to Dr. Albin Brandstetter.

REQUEST FOR MODEL REPORT REVISION

Name_____

Title and Affiliation_____

Address_____

Date_____ Signature_____

Please fill out this form and mail to:

Dr. Albin Brandstetter
Project Manager, AEGIS
Battelle, Pacific Northwest Laboratory
P.O. Box 999
Richland, WA 99352
U.S.A.