START

0011611

PNL-3160-3
UC-70

BEST AVAILABLE COPY

Assessment of Effectiveness of Geologic Isolation Systems

VARIABLE THICKNESS TRANSIENT GROUND-WATER FLOW MODEL

VOLUME 3. PROGRAM LISTINGS

A. E. Reisenauer

December 1979

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

THIS PAGE INTENTIONALLY
LEFT BLANK

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.

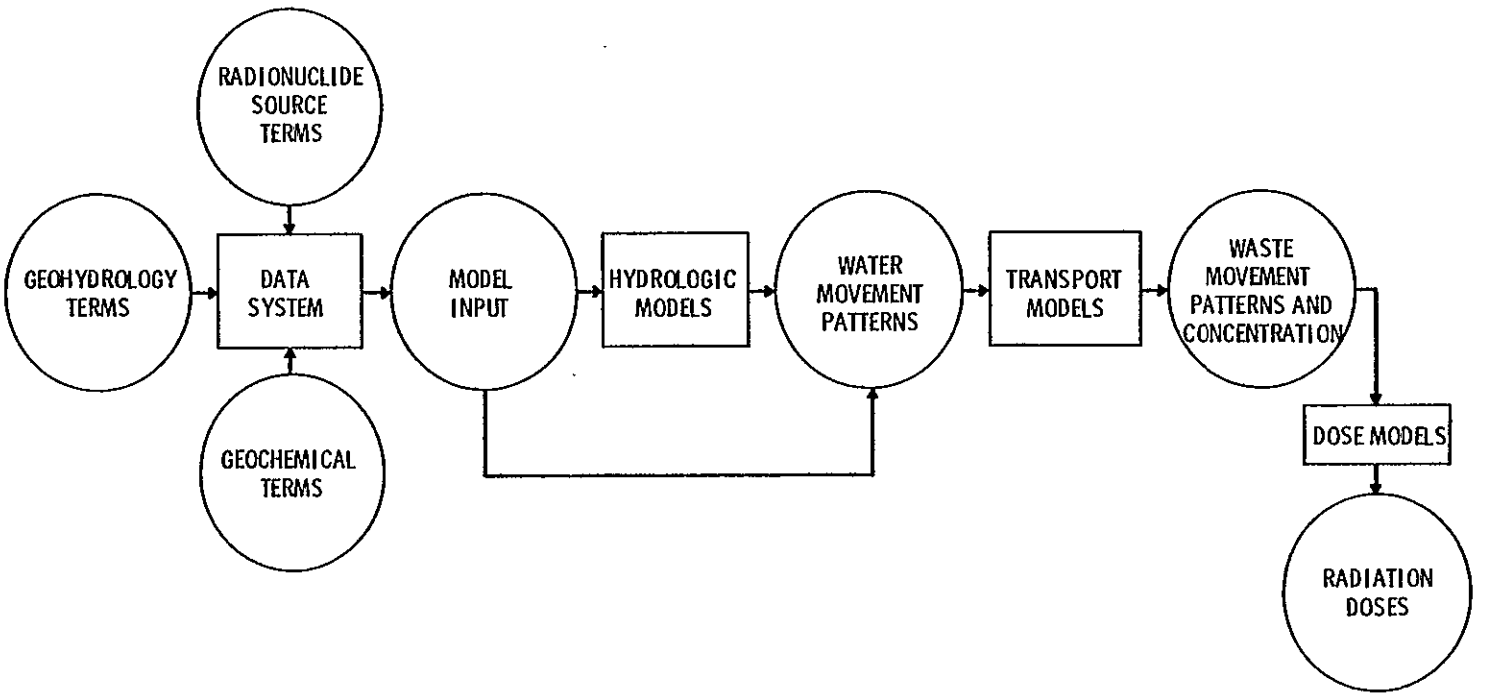

FIGURE i. Schematic Diagram of Consequence Analysis

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-member 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 third of 3 volumes of the description of the VTT hydrologic model.

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

FOREWORD	iii
ACKNOWLEDGMENTS	vii
VTTINP	1
VTT	15
VTTSSX	45
VTTSS3	69
ITTER	93
READN	97
MXED	121
SUBGEN	125
BCTRN	139
LISTP	145
ACHEK	151
MXPLT	157
CALTYP	183
DIFER	191
DAYQT	201
WIQPG	207
WISAPQ	211

```
DB0:VTTINP,LP/SH=VTTINP/CC,TIME/CC
[1,1]BNWLIB/LB
/
ASG=TI: 6,DB0:2,DP0:1:3:4:5
UNITS=7
ACTFIL=6
/
```

```
        C ******** VTTINP **********
        C ******** MAIN INPUT PROGRAM FOR THE VTT GROUNDWATER MODEL
        C
        C
        C VTTINP,LP/SH=VTTINP/CC,TIME/CC
        C [1,1]FLQLIB.OLB/LB
        C ICHR
        C /
        C ASG=TI: 6,DP2: 2,DP0: 1:3:4:5
        C UNITS=7
        C ACTFIL=6
        C /
        C
0001            BYTE STAT,DOIT,DOIT1,TITLE
0002            INTEGER TIMVAL,QTIME,QPTR
0003            DIMENSION POT(7),BOT(7),TOP(7),TRN(7),CAL(7),STO(7),TRCF(7)
                1,OTRCF(7),FILIN(7),POTIC(7),RLEAK(7)
0004            DIMENSION NODSTY(2,128),NODLOC(256)
0005            DIMENSION IOUT(256),ARG(4),NTP(2,128),NTP1(2,128)
0006            DIMENSION BOTSL(128),TRNSL(128),TOPSL(128),POTSL(128)
0007            DIMENSION TOUT(128),BOUT(128),QSL0(128),QSL1(128),NOTYP(256)
0008            REAL NONE
0009            COMMON /INP/A(6000),OUT(128),NAQS,NODESX,NODESY
                1,IROT,BIAS,IPTR,IPTR2,IPTR3,IPTR4
0010            COMMON /TIMR/ NQTS,NTPLNS,TIMVAL(5,25),RATIO(25),NSTEPS(25)
                1,QTIME(5,28),QPTR(28),IFIN(142)
0011            COMMON /BLK0/NUNK,IHOW,NLNS,NCLS,NITT,ITTCNT,CONV,
                1BIAS,NQS,IOXFER,IRT,MAXIT,DELX,IAUTO,OMEGA,
                2NTMPLN,NQTIMS,ITROSS,IAOW,TIMEC,STIME,ETIME,
                3IYEAR,IMON,IDAY,IHOUR,IMIN,IYRL,IMONL,IDAYL,IHRL,IMINL,
                4NDSX,NDSY,NAQFNS,NUCONF,DOIT,DOIT1,STOMUL,TITLE(80)
                5,COFLEK,SEALEV,DELLEK,ITVN,ILRGSM,
                1ISMLSM,XLLH,YLLH,FAKE(81)
0012            EQUIVALENCE (A(1),BOTSL(1))
0013            EQUIVALENCE (A(257),TOPSL(1))
0014            EQUIVALENCE (A(513),POTSL(1))
0015            EQUIVALENCE (A(769),TOUT(1))
0016            EQUIVALENCE (A(1025),BOUT(1))
0017            EQUIVALENCE (A(1281),QSL0(1))
0018            EQUIVALENCE (A(1537),QSL1(1))
0019            EQUIVALENCE (A(1793),NTP(1,1))
0020            EQUIVALENCE (A(2049),NTP1(1,1))
0021            EQUIVALENCE (A(2305),NODSTY(1,1))
0022            EQUIVALENCE (A(2561),NODLOC(1))
0023            EQUIVALENCE (A(2817),TRNSL(1))
0024            EQUIVALENCE (A(3073),NOTYP(1))
0025            EQUIVALENCE (A(3329),IOUT(1))
        C
0026            DATA ARG/1.,0.5,.375,.625/
0027            DATA NONE/'NONE'/
        C
0028            WRITE(6,15)
0029    15      FORMAT(' WHICH FILE O DISK 1,2,3=DBO:')
0030            READ(6,20) IUN
0031    20      FORMAT(I5)
0032            IF(IUN.EQ.0) IUN=1
```

```
0033            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,STAT)
0034            DOIT1=0
0035            WRITE(6,8899)
0036    8899    FORMAT(' HAVE YOU RUN QPRG WITH THE PROPER ROTATION????')
0037            WRITE(6,100)
0038    100     FORMAT(' ENTER INPUT FILE NAME')
0039            READ(6,2)FILIN
0040            N=ICHR(FILIN)
0041            WRITE(6,101)FILIN
0042    101     FORMAT(1X,7A4)
0043            ILRGSM=0
0044    8798    FORMAT(I5)
0045            WRITE(6,8797)
0046    8797    FORMAT(' TIME VARING BOUNDARY NODES (0=NO, 1=YES)')
0047            READ(6,8798) ITVN
0048            WRITE(6,8796)
0049    8796    FORMAT(' IS A SMALL REGION SIMULATION TO BE RUN '
                1,'(0=NO 1=YES)')
0050            READ(6,8798) ISMLSM
0051            IQAD1=0.
0052            IQAD2=0.
0053            IF(ILRGSM.EQ.1) IQAD1=5601
0054            IF(ILRGSM.EQ.1) IQAD2=2.
0055            WRITE(6,8795)
0056    8795    FORMAT(' ENTER NAME OF INITIAL CONDITION FILE(OR NONE)')
0057            READ(6,8794) POTIC
0058    8794    FORMAT(7A4)
0059            IF(ILRGSM.EQ.1) IQAD2=2.
0060            CLOSE(UNIT=6)
0061            CALL ASNLUN(6,'DP',0,ISTAT)
0062            OPEN(UNIT=6,NAME=FILIN,TYPE='OLD',READONLY)
0063            READ(6,2135) TITLE
0064    2135    FORMAT(80A1)
0065            READ(6,2) POT,BOT,TOP,TRN,CAL,STO,TRCF,OTRCF
0066    2       FORMAT(7A4)
0067            N=ICHR(POT)
0068            N=ICHR(BOT)
0069            N=ICHR(TOP)
0070            N=ICHR(TRN)
0071            N=ICHR(CAL)
0072            N=ICHR(STO)
0073            N=ICHR(TRCF)
0074            N=ICHR(OTRCF)
0075            READ(6,4) NAQS,NODESX,NODESY,DOIT,DOIT1,IROT,BIAS,DELX
                1,STOMUL
0076    4       FORMAT(3I5,2A1,I2,3F10.0)
0077            READ(6,6000) XLLH,YLLH
0078    6000    FORMAT(2F10.0)
0079            IF(DOIT1.EQ.'Y') GO TO 1111
0080            DOIT1=0
0081            COFLEK=0.
0082            READ(6,1112) SEALEV
0083    1112    FORMAT(3F10.0)
0084            GO TO 1113
0085    1111    DOIT1=1
0086            READ(6,11120) SEALEV,COFLEK,DELLEK,RLEAK
```

```
0087            N=ICHR(RLEAK)
0088            COFLEK=DELX*DELX*COFLEK
0089    11120   FORMAT(3F10.0,7A4)
0090    1113    CONTINUE
0091            SEALEV=SEALEV+BIAS
0092            NDSX=NODESX
0093            NDSY=NODESY
0094            NAOFRS=NAWS
0095            BIASS=BIAS
0096            NOS=NAOS
0097            IRT=IROT
0098            CALL OPFILE(2,'VTTQS',11205.,IQPTR)
0099            CALL OPR(IQPTR,11201.+IQAD2,NQTS,512.)
0100            READ(6,5) ITROSS,IOXFER,IAUTO,NITT,MAXIT,CONV,OMEGA
0101    5       FORMAT(5I5,2F10.0)
0102            READ(6,6)IYEAR,IMON,IDAY,IHOUR,IMIN
0103    6       FORMAT(5I5)
0104            READ(6,6)IYRL,IMONL,IDAYL,IHRL,IMINL
        C WHICH MODE OF TIME STEP INPUT
0105            READ(6,6555) IKND,NTMPLN,TOTM,TINT
0106    6555    FORMAT(2I5,2F10.0)
0107            IF(IKND.EN.0) GO TO 6556
0108            READ(6,6557) (NSTEPS(I),I=1,NTMPLN)
0109    6557    FORMAT(20I3)
0110            TIMVAL(1,1)=IYEAR
0111            TIMVAL(2,1)=IMON
0112            TIMVAL(3,1)=IDAY
0113            TIMVAL(4,1)=IHOUR
0114            TIMVAL(5,1)=IMIN
0115            CALL TIMER(TIMVAL,NTMPLN,TOTM,TINT,R)
0116            DO 6558 I=1,NTMPLN
0117            RATIO(I)=R
0118            IF(NSTEPS(I).EQ.0) NSTEPS(I)=1
0119    6558    CONTINUE
0120            TIMVAL(1,NTMPLN)=IYRL
0121            TIMVAL(2,NTMPLN)=IMONL
0122            TIMVAL(3,NTMPLN)=IDAYL
0123            TIMVAL(4,NTMPLN)=IHRL
0124            TIMVAL(5,NTMPLN)=IMINL
0125            GO TO 7777
0126    6556    DO 7 ITM=1,NTMPLN
0127            READ(6,8) (TIMVAL(J,ITM),J=1,5),RATIO(ITM),NSTEPS(ITM)
0128    8       FORMAT(5I5,F10.0,I5)
0129    7       CONTINUE
0130    7777    NTPLNS=NTMPLN
0131            CALL OPW(IQPTR,11201.+IQAD2,NQTS,512.)
        C SET UP THE DISK FILE
0132            CALL OPFILE(2,'VTTFIELDS',2401.,IPTR)
0133            IF(ILRGSM.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,IPTR)
0134            CALL OPFILE(2,'BOTTOMTRN',400.,IPTR2)
0135            CALL OPFILE(2,'VTTTEMP0',400.,IPTR3)
0136            CALL OPFILE(2,'VTTTEMP1',400.,IPTR4)
        C INPUT POTENTIALS AND OUTPUT TO FILE ROTATING IF NECESSARY
0137            CALL GETEM(POT,IPTR,1.,12.,BIAS,1.,1)
        C NOW DO THE BOTTOMS
0138            CALL GETEM(BOT,IPTR2,1.,2.,BIAS,1.,2)
```

```
        C NOW DO THE TOPS
0139            CALL GETEM(TOP,IPTR,7.,12.,BIAS,1.,3)
        C NOW DO THE TRANSMISSIVITY
0140            CALL GETEM(TRN,IPTR2,2.,2.,0.,(1./7.481),4)
        C NOW DO THE CALCULATIONAL TYPES
0141            CALL GETEM(CAL,IPTR,5.,12.,0.,1.,5)
        C NOW DO THE STORAGE COEFFICIENTS
0142            CALL GETEM(STO,IPTR,11.,12.,0.,1.,6)
        C NOW DO THE INTERAQUIFER TRANSFER COEFFICIENTS
0143            CALL GETEM(TRCF,IPTR,8.,12.,0.,1.,7)
        C NOW DO THE OCEAN TRANSFER COEFFICIENTS
0144            CALL GETEM(OTRCF,IPTR,9.,12.,0.,1.,8)
        C       ALL CALCULATIONAL TYPES AVAILABLE
0145            IF(IROT.EQ.1) GO TO 701
        C NO ROTATION
0146            NY=NODESY
0147            NX=NODESX*NAQS
0148            NLNS=NY
0149            NCLS=NX
0150            IADW=NODESX
0151            GO TO 702
        C ROTATION
0152    701     NX=NODESY*NAQS
0153            NY=NODESX
0154            NCLS=NX
0155            NLNS=NY
0156            IADW=NODESY
0157    702     CONTINUE
0158            NMAX=0
0159            NOACT=0
0160            DO 60 ILIN=1,NY
0161            NOPER=0
0162            RB=(ILIN-1)*12.+5.
0163            CALL DPR(IPTR,RB,NDTYP,256.)
0164            DO 61 I=1,128
0165            NODLOC(I*2)=0
0166            NODLOC(I*2-1)=0
0167            NODSTY(1,I)=0
0168            NODSTY(2,I)=0
0169    61      CONTINUE
0170            DO 616 ICOL=1,NX
0171            ITYP=NDTYP(ICOL)
0172            NODSTY(2,ICOL)=ITYP
0173            IF(ITYP.EQ.1.OR.ITYP.EQ.2) GO TO 616
0174            NOACT=NOACT+1
0175            NOPER=NOPER+1
0176            NLOC=NOPER+1
0177            NODSTY(1,ICOL)=NOACT
0178            NODLOC(NLOC)=ICOL
0179    616     CONTINUE
0180            IF(NOPER.GT.NMAX) NMAX=NOPER
0181            NODLOC(1)=NOPER
0182            CALL DPW(IPTR,RB,NODSTY,256.)
0183            CALL DPW(IPTR,RB+1.,NODLOC,256.)
0184    60      CONTINUE
        C CALCULATE THE HALFBAND WIDTH
```

```
0185            IMAX=0
0186            DO 400 ILN=1,NY-1
0187            RB=(ILN-1)*12.+5.
0188            RB1=RB+12.
0189            CALL DPR(IPTR,RB,NTP,256.,STAT)
0190            CALL DPR(IPTR,RB1,NTP1,256.,STAT)
0191            DO 400 IP=1,NX
0192            IN0=NTP(1,IP)
0193            IN1=NTP1(1,IP)
0194            IF(IN0.EQ.0.OR.IN1.EQ.0) GO TO 400
0195            IDF=IN1-IN0
0196            IF(IDF.GT.IMAX) IMAX=IDF
0197    400     CONTINUE
0198            IHBW=IMAX
        C DO THE AREA AND NODE TYPE ADJUSTMENTS TO THE TRANSFER COEFFICIENTS
0199            DELX2=DELX**2
0200            DO 92 ILIN=1,NY
0201            RB=(ILIN-1)*12.+5.
0202            RB1=RB+3.
0203            RB2=RB+4.
0204            CALL DPR(IPTR,RB,IOUT,256.)
0205            CALL DPR(IPTR,RB1,TRNSL,256.)
0206            CALL DPR(IPTR,RB2,TOPSL,256.)
0207            DO 94 IAQ=1,NAQS
0208            DO 94 IWO=1,IAQW
0209            ICOL=IWO+(IAQ-1)*IAQW
0210            IFRM=ICOL+IAQW
0211            IF(IAQ.EQ.3) IFRM=ICOL-2*IAQW
0212            IF(IAQ.EQ.2) IFRM=ICOL+IAQW
0213            NFRM=IOUT(2*IFRM)
0214            NNOD=IOUT(2*ICOL)
0215            ARF=ARG(NNOD/8+1)
0216            ARFF=ARG(NFRM/8+1)
0217            IF(NFRM.EQ.1) ARFF=0.
0218            IF(NNOD.EQ.1) ARF=0.
0219            ARFF=MIN(ARFF,ARF)
0220            TRNSL(ICOL)=TRNSL(ICOL)*ARFF*DELX2
0221            TOPSL(ICOL)=TOPSL(ICOL)*ARF*DELX2
0222    94      CONTINUE
0223            CALL DPW(IPTR,RB1,TRNSL,256.)
0224            CALL DPW(IPTR,RB2,TOPSL,256.)
0225    92      CONTINUE
        C INPUT THE Q'S NOW
0226            IQ=1
0227            DO 95 I=1,NLNS
0228            QADDR1=200.*(IQ-1)+I+IQAD1
0229            QADDR2=QADDR1+200.
0230            CALL DPR(IQPTR,QADDR1,QSL0,256.)
0231            CALL DPR(IQPTR,QADDR2,QSL1,256.)
0232            RB1=(I-1)*12.+4.
0233            RB2=RB1+8.
0234            CALL DPW(IPTR,RB1,QSL0,256.)
0235            CALL DPW(IPTR,RB2,QSL1,256.)
0236    95      CONTINUE
        C----BEGIN READING DATA FOR CALCULATION------
0237            NUCONF=0
```

```
0238            DO 1000 L=1,NY
0239            BLK=(L-1)*12.
0240            CALL DPR(IPTR,BLK+5.,NTP,256.)
0241            CALL DPR(IPTR,BLK+1.,POTSL,256.)
0242            CALL DPR(IPTR,BLK+7.,TOPSL,256.)
0243            DSEC=(L-1)*2.+1
0244            CALL DPR(IPTR2,DSEC,BOTSL,256.)
0245            CALL DPR(IPTR2,DSEC+1.,TRNSL,256.)
        C----CALCULATION LOOP------
        C
        C       CONFINED?
0246            DO 4000 I=1,NX
0247            IF(TOPSL(I)-POTSL(I))2000,2500,3000
        C----CONFINED AQUIFER-----
0248    2000    THICK=TOPSL(I)-BOTSL(I)
0249            IF(THICK.LE.0) BOTSL(I)=TOPSL(I)-25.0
0250            IF(THICK.LE.0) THICK=25.0
0251            TOUT(I)=ABS(TRNSL(I))/THICK
0252            IF(TRNSL(I).GE.0.) TOUT(I)=TRNSL(I)
0253            IF(DOIT.EQ.'Y') GO TO 2100
0254            TOUT(I)=-TOUT(I)
0255            BOUT(I)=THICK
0256            GO TO 4000
0257    2100    BOUT(I)=BOTSL(I)
0258            GO TO 4000
        C----OUTSIDE AQUIFER------
0259    2500    IF(NTP(2,I).NE.1) GO TO 2000
0260            TOUT(I)=0.0
0261            BOUT(I)=0.0
0262            GO TO 4000
        C-----UNCONFINED AQUIFER-------
0263    3000    THICK=POTSL(I)-BOTSL(I)
0264            IF(THICK.LE.0) BOTSL(I)=POTSL(I)-25.0
0265            IF(THICK.LE.0) THICK=25.0
0266            TOUT(I)=ABS(TRNSL(I))/THICK
0267            IF(TRNSL(I).GE.0.) TOUT(I)=TRNSL(I)
0268            BOUT(I)=BOTSL(I)
0269            NUCONF=NUCONF+1
0270            IF(DOIT.EQ.'Y') GO TO 4000
0271            NUCONF=NUCONF-1
0272            TOUT(I)=-ABS(TOUT(I))
0273            BOUT(I)=THICK
0274    4000    CONTINUE
0275            CALL DPW(IPTR,BLK+2.,BOUT,256.)
0276            CALL DPW(IPTR,BLK+3.,TOUT,256.)
0277    1000    CONTINUE
        C       WRITE(7,5555) NUCONF
        C5555   FORMAT(' NUMBER OF UNCONFINED NODES =',I7)
0278            CLOSE(UNIT=1)
        C----WRITE HEADER BLOCK------
0279            NUNK=NOACT
0280            IF(ITROSS.EQ.1) NITT=4
0281            ITTCNT=0
0282            TIMEC=0.
0283            STIME=0.
0284            ETIME=0.
```

```
FORTRAN IV-PLUS V02-51D          16:14:08     30-MAY-79
VTTINP.FTN       /TR:8LOCKS/WR

0285             CALL DPW(IPTR,0.,NUNK,256.)
       C SET UP INITIAL CONDITIONS AND LEAKAGE POTENTIALS IF REQUIRED
0286             IF(DOIT1.NE.1) GO TO 8347
0287             CALL GETEM(RLEAK,IPTR,8.,12.,BIAS,1.,1)
0288     8347    CONTINUE
0289             IF(POTIC(1).EQ.NONE) STOP
0290             N=ICHR(POTIC)
0291             CALL GETEM(POTIC,IPTR,1.,12.,BIAS,1.,1)
0292             STOP
0293             END
```

```
0001            SUBROUTINE GETEN(FILE,IFLG,POS,SIZ,BIAS,SFACT,IKND)
0002            REAL NONE
0003            DIMENSION ICOLG(2,128),IA(12000),IOT(256),FILE(7),VAL(200),IVAL(400)
0004            COMMON /GET/TRNSL(128),IOUT(256)
0005            COMMON /INP/A(6000),OUT(128),NAQS,NODESX,NODESY
                1,IROT,BSSS,IPTR,IPTR2,IPTR3,IPTR4
0006            EQUIVALENCE (TRNSL(1),ICOLG(1,1))
0007            EQUIVALENCE (A(1),IA(1)),(VAL(1),IVAL(1))
0008            EQUIVALENCE (OUT(1),IOT(1))
0009            DATA NONE/'NONE'/
0010            IODD=0
0011            DO 10 IAQ=1,NAQS
        C READ IN THE UNROTATED ARRAY FOR THE CURRENT AQUIFER
0012            IUN=1
0013            IF(FILE(1).NE.NONE) GO TO 1
        C FILE NAME IS NONE SO INPUT IS IN MAIN INPUT STREAM
0014            IUN=6
0015            IF(IKND.EQ.5) GO TO 2
        C NORMAL COLUMN OF CHANGE INPUT FROM LUN IUN
        C EVERY VARIABLE TYPE BUT CALCULATIONAL TYPES
0016            IODD=1
0017    101     READ(IUN,51) IB,IT,KKK
0018    51      FORMAT(3I3)
0019            READ(IUN,36) (IOUT(I),TRNSL(I),I=1,KKK)
0020    36      FORMAT(5(I5,F10.0))
0021            DO 360 ICHNG=1,KKK
0022            IS1=IOUT(ICHNG)
0023            IS2=IOUT(ICHNG+1)
0024            IF((ICHNG+1).GT.KKK) IS2=NODESX
0025            VAL1=TRNSL(ICHNG)
0026            DO 360 ICOL=IS1,IS2
0027            VAL(ICOL)=VAL1
0028    360     CONTINUE
0029            DO 361 ILIN=IB,IT
0030            ADR=(ILIN-1)*2.
0031            CALL DPW(IPTR4,ADR,VAL,NODESX*2.)
0032    361     CONTINUE
0033    362     CONTINUE
0034            IF(IT.LT.NODESY) GO TO 101
0035            GO TO 30
        C NORMAL INPUT FROM LUN 2
0036    1       CONTINUE
0037            IUN=1
0038            IF(IKND.NE.5) GO TO 3
0039            IF(IAQ.EQ.1) OPEN(UNIT=1,NAME=FILE,TYPE='OLD',FORM=
                1'FORMATTED',READONLY)
0040            GO TO 2
0041    3       CONTINUE
0042            IF(IAQ.EQ.1) OPEN(UNIT=1,NAME=FILE,TYPE='OLD',FORM=
                1'UNFORMATTED',READONLY)
0043    30      NLIN=6000/NODESX
0044            IF(NLIN.GT.NODESY)NLIN=NODESY
0045            NPGS=NODESY/NLIN
0046            IF(MOD(NODESY,NLIN).NE.0)NPGS=NPGS+1
0047            DO 4 IPG=1,NPGS
```

```
0048            ISTLN=(IPG-1)*NLIN+1
0049            ISTPLN=ISTLN-1+NLIN
0050            IF(ISTPLN.GT.NODESY)ISTPLN=NODESY
0051            DO 44 ILIN=1,ISTPLN-ISTLN+1
0052            ADR=(ISTLN+ILIN-2)*2.
0053            IDX1=(ILIN-1)*NODESX+1
0054            IDX2=ILIN*NODESX
0055            IF(IODD.EQ.0)READ(1)(A(I),I=IDX1,IDX2)
0056            IF(IODD.EQ.1)CALL DPR(IPTR4,ADR,A(IDX1),NODESX*2.)
0057            IF(IODD.EQ.2)CALL DPR(IPTR4,ADR,IA(IDX1),RNWD)
0058    44      CONTINUE
        C ROTATION REQUIRED?
0059            IF(IROT.NE.1)GO TO 444
        C YES ROTATE
0060            WORDS=(NODESY*2.)*NAQS
0061            DO 45 ILIN=1,NODESX
0062            ADR=(ILIN-1)*2.
0063            CALL DPR(IPTR3,ADR,VAL,WORDS)
0064            DO 46 ICOL=ISTPLN,ISTLN,-1
        C       K=(ICOL-1)*NODESX+ILIN
0065            K=(ICOL-ISTLN)*NODESX+ILIN
        C       J=(IAQ-1)*NODESY+ICOL
0066            J=(IAQ-1)*NODESY+NODESY-ICOL+1
0067            IF(IODD.NE.2)VAL(J)=A(K)
0068            IF(IODD.EQ.2)IVAL(J)=IA(K)
0069            IF(IODD.NE.2.AND.A(K).NE.0)VAL(J)=(A(K)+BIAS)*SFACT
0070    46      CONTINUE
0071            CALL DPW(IPTR3,ADR,VAL,WORDS)
0072    45      CONTINUE
0073            GO TO 4
        C NO ROTATION
0074    444     CONTINUE
0075            WORDS=(NODESX*2.)*NAQS
0076            DO 445 ILIN=ISTLN,ISTPLN
0077            L=(ILIN-ISTLN)*NODESX
0078            ADR=(ILIN-1)*2.
0079            CALL DPR(IPTR5,ADR,VAL,WORDS)
0080            DO 446 J=(IAQ-1)*NODESX+1,IAQ*NODESX
0081            K=J-(IAQ-1)*NODESX+L
0082            IF(IODD.NE.2)VAL(J)=A(K)
0083            IF(IODD.EQ.2)IVAL(J)=IA(K)
0084            IF(IODD.NE.2.AND.A(K).NE.0)VAL(J)=(A(K)+BIAS)*SFACT
0085    446     CONTINUE
0086            CALL DPW(IPTR3,ADR,VAL,WORDS)
0087    445     CONTINUE
0088    4       CONTINUE
0089            GO TO 10
        C COLUMN OF CHANGE INPUT OF CALCULATIONAL TYPES FROM LUN IUN
0090    2       CONTINUE
0091            IF(IUN.EQ.6) GO TO 20
0092            READ(IUN,51)
0093            READ(IUN,51)
0094            READ(IUN,51)
0095    20      CONTINUE
0096            IODD=2
0097    21      READ(IUN,51) IB,IT,KKK
```

```
0098            READ(IUN,52) ((ICOLG(I,JJJ),I=1,2),JJJ=1,KKK)
0099     52     FORMAT(14(I3,O2))
0100            DO 53 ICHNG=1,KKK
0101            IS1=ICOLG(1,ICHNG)
0102            IKIND=ICOLG(2,ICHNG)
0103            KKIND=IKIND
0104            IF(IROT.EQ.1) CALL ROTATE(IKIND,KKIND)
0105            IS2=ICOLG(1,ICHNG+1)-1
0106            IF(ICHNG+1.GT.KKK) IS2=NODESX
0107            DO 54 ICOL=IS1,IS2
0108     54     IVAL(ICOL)=KKIND
0109     53     CONTINUE
0110            RNWD=((NODESX+1)/2)*2.
0111            DO 531 ILIN=IB,IT
0112            ADR=(ILIN-1.)*2.
0113            CALL DPW(IPTR4,ADR,IVAL,RNWD)
0114     531    CONTINUE
0115            IF(IT.LT.NODESY) GO TO 20
0116            GO TO 30
0117     10     CONTINUE
         C COPY BOTTOMTRN TO VTTFIELDS
0118            NLS=NODESY
0119            NCL=NODESX*NAQS
0120            IF(IROT.EQ.0)GO TO 11
0121            NLS=NODESX
0122            NCL=NODESY*NAQS
0123     11     CONTINUE
0124            DO 12 IL=1,NLS
0125            ADR1=(IL-1)*2.
0126            ADR2=(IL-1)*SIZ+POS
0127            CALL DPR(IPTR3,ADR1,VAL,NCL*2.)
0128            CALL DPW(IFLG,ADR2,VAL,NCL*2.)
0129     12     CONTINUE
0130            IF(IUN.EQ.6) RETURN
0131            CLOSE(UNIT=IUN)
0132            RETURN
0133            END
```

```
FORTRAN IV-PLUS V02-510          16:16:05     30-MAY-79
VTTINP.FTN       /TR:BLOCKS/WR

0001             SUBROUTINE ROTATE(ITYPE,KTYPE)
0002             IF(ITYPE-3)99,99,1
0003     1       IROT=IFLO(0,3,ITYPE)
0004             IF(IROT.EQ.0)GO TO 5
0005             IF(IROT.EQ.1)GO TO 10
0006             IROT=IROT-2
0007             GO TO 20
0008     5       IROT=6
0009             GO TO 20
0010     10      IROT=7
0011     20      CALL SFLO(0,3,IROT,ITYPE)
0012     99      KTYPE=ITYPE
0013             RETURN
0014             END
```

```
FORTRAN IV-PLUS V02-51D          16:16:17     30-MAY-79
TIME.FTN          /TR:BLOCKS/WR

        C PROGRAM TO GENERATE TIME STEPS
0001              SUBROUTINE TIMER(TIMVAL,NSTP,TOTIM,TIMINT,R)
0002              INTEGER TIMVAL(5,20)
0003              DIMENSION MON(12)
0004              DATA MON/31,28,31,30,31,30,31,31,30,31,30,31/
0005    1000      CONTINUE
        D         WRITE(6,124)
        D124      FORMAT(' ENTER THE INITIAL TIME STEP,TOTAL TIME,NO. STEPS'
        D         1/' STARTING YEAR,MONTH,DAY,HOUR,MINUTE')
        D         READ(6,123) TIMINT,TOTIM,NSTP
        D123      FORMAT(2F15.3,I8)
        D         READ (6,125) IYEAR,IMON,IDAY,IHOUR,IMIN
        D125      FORMAT(5I7)
0006              P=RATE(TIMINT,TOTIM,NSTP)
0007              IYEAR=TIMVAL(1,1)
0008              IMON=TIMVAL(2,1)
0009              IDAY=TIMVAL(3,1)
0010              IHOUR=TIMVAL(4,1)
0011              IMIN=TIMVAL(5,1)
0012              IY=IYEAR
0013              IM=IMON
0014              ID=IDAY
0015              HRS=IHOUR
0016              RMNS=IMIN
0017              TSTP=TIMINT/R
0018              DO 3 I=1,NSTP
0019              TSTP=TSTP*R
0020              RMNS=RMNS+TSTP
0021              HRS=HRS+IFIX(RMNS/60.)
0022              RMNS=RMNS-IFIX(RMNS/60.)*60.
0023              ID=ID+IFIX(HRS/24.)
0024              HRS=HRS-IFIX(HRS/24.)*24.
0025              IDD=MON(IM)
0026              IF(IM.EQ.2.AND.MOD(IY,4).EQ.0)IDD=IDD+1
0027              IF(ID.LT.IDD) GO TO 1
0028              ID=ID-IDD
0029              IM=IM+1
0030              IF(IM.LT.13) GO TO 1
0031              IY=IY+1
0032              IM=1
0033    1         CONTINUE
0034              TIMVAL(1,I)=IY
0035              TIMVAL(2,I)=IM
0036              TIMVAL(3,I)=ID
0037              TIMVAL(4,I)=HRS
0038              TIMVAL(5,I)=RMNS+.5
        D         WRITE(6,2) I,R,TSTP,(TIMVAL(J,I),J=1,5)
        D2        FORMAT(I5,2F15.7,5I5)
0039    3         CONTINUE
0040              RETURN
0041              END

FORTRAN IV-PLUS V02-51D          16:16:31     30-MAY-79          PAGE 4
TIME.FTN          /TR:BLOCKS/WR

0001              FUNCTION FSUM(T,R,N)
0002              FSUM=T/R**(N-1)*(1.-R**N)/(1.-R)
0003              RETURN
0004              END
```

```
FORTRAN IV-PLUS V02-510          16:16:26     30-MAY-79
TIME.FTN          /TR:BLOCKS/WR

0001              FUNCTION RATE(TIMINT,TOTIM,NSTP)
0002              TOTTM=TOTIM*1440.
0003              R0=1.01
0004              R1=2.0
0005              S0=FSUM(TIMINT,1./R0,NSTP)
0006              S1=FSUM(TIMINT,1./R1,NSTP)
0007              SD0=TOTTM-S0
0008              SD1=TOTTM-S1
0009      16      CONTINUE
0010              IF(ABS(SD1).LE.1.0) GO TO 13
0011              IF(ABS(SD0).GT.1.0) GO TO 14
0012              SD1=SD0
0013              R1=R0
0014              S1=S0
0015              GO TO 13
0016      14      RX=R0+(R1-R0)/2.
0017              SX=FSUM(TIMINT,1./RX,NSTP)
0018              SDX=TOTTM-SX
0019              IF(SDX.LT.0.) GO TO 15
          C SDX GT 0
          C REPLACE R0
0020              R0=RX
0021              S0=SX
0022              SD0=SDX
0023              GO TO 16
          C SDX LT 0
          C REPLACE R1
0024      15      R1=RX
0025              S1=SX
0026              SD1=SDX
0027              GO TO 16
0028      13      CONTINUE
0029              RATE=R1
0030              RETURN
0031              END
```

```
DB0:VTT,LP:/SH=VTT/CC
[1,1]BNWLIB/LB
TRIDIG
/
ASG=DB0:2
ASG=LP0: 5
ASG=TI:6
/
```

```
FORTRAN IV-PLUS V02-510          16:16:42     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

      C ******* VTT (TRANSIENT VERSION) WITH NEW IMPROVEMENTS
      C ******* MAIN CALCULATIONAL PROGRAM FOR THE TRANSIENT VTT SIMULATION
      C VTT/-CP,LP:/SH=VTT/CC
      C [1,1]8NWLIB,OLB/LB
      C TRIDIG
      C /
      C ASG=DB0:2
      C ASG=LP0: 5
      C ASG=TI:6
      C /
0001            BYTE DOIT,DOIT1,TITLE,FSTAT
0002            INTEGER TIMVAL,QPTR,QTIME,RTIMES
0003            INTEGER OXFER,TRNOSS,YEAR,DAY,HOUR,YEARL,DAYL,HOURL,AUTO
0004            REAL K4,K0,K2,K,NWDS
0005            DOUBLE PRECISION STIME,TIME,ETIME,QTIME0,QTIME1,TPLANE,
                1DTOTIM,DT1,DTFR,TIME0,QTIM,QTIMP,ATIME,ACTIM
0006            DIMENSION BLK0(100),FA(4608)
0007            DIMENSION ACOEF(4,27),IG(512),AREA(4),INDX3(4,3),ITIMC(5)
      C COMMON BLOCK CONSISTS OF 248. WORDS
0008            COMMON/ADDWS/IDT1M0(5),NWLLSL,NWLLSS,WLRATL(30),IWLOCL(2,30)
                1,WLRATS(30),IWLOCS(2,30),LOST
      C COMMON BLOCK EQUIVALENCED TO FA(4608) 9216 WORDS
0009            COMMON /ALL/
                1H2(128),B2(128),K2(128),Q2(128),NTYP2(256),
                2NPOS2(256),TOP2(128),COEF2(128),OCOEF2(128),
                3OH2(128),STO2(128),QNEX2(128),
                4H0(128),B0(128),K0(126),Q0(128),NTYP0(256),
                5NPOS0(256),TOP0(128),COEF0(128),OCOEF0(128),
                6OH0(128),STO0(128),QNEX0(128),
                7H4(128),B4(128),K4(128),Q4(128),NTYP4(256),
                8NPOS4(256),TOP4(128),COEF4(128),OCOEF4(128),
                9OH4(128),STO4(128),QNEX4(128)
      C HEADER COMMON BLOCK 100 WORDS
0010            COMMON /BK0/ NUNK,IHBW,NLINES,NCOLS,NITT,ITTCNT,CONV,
                1BIAS,NAQS,OXFER,IROT,MAXIT,DELX,AUTO,OMEGA,NTMPLN,
                2NQTIMS,TRNOSS,IWID,TIMEE,STIMEE,ETIMEE,YEAR,MONTH,DAY,
                3HOUR,MIN,YEARL,MONTHL,DAYL,HOURL,MINL,NOSX,NDSY,NAQFRS,
                4NUCONF,DOIT,DOIT1,STOMUL,TITLE(80),
                5COFLEK,SEALEV,DELLEK,ITVN,ILRGSM,ISMLSM,
                6XLLH,YLLH,IFAKE
      C COMMON BLOCK FOR THE LOCAL SUBROUTINE
0011            COMMON H(7),HP(7),T(5),TP(5),K(5),BOT(5),TOP(5),S,C6,C7,CO,
                1A0(4),Q,QP,QTIM,QTIMP,ATIME,DTIME,DLX2,T0(4),OCEAN,STOFCT
                1,POTOIF
      C COMMON BLOCK FOR THE TRIDIAGONAL SUBROUTINE
0012            COMMON /TRIDG/ A(128),B(128),C(128),P(128),Z(128)
      C COMMON BLOCK FOR THE RESULT FILE 602 WORDS
0013            COMMON /RESULT/NUMSAV,RTIMES(3,100),RMIN(100),NTPLSV,ISAVNO(100)
      C COMMON BLOCK FOR FILE-Q POINTERS
0014            COMMON /PTRS/IPTR,IQPTR,IANS,IRSPTR,ISTATS,IADQ
      C COMMON BLOCK FOR THE RESTART FILE 60 WORDS
0015            COMMON /RESTRT/LTMPLN,STIME,ETIME,TIME,TIME0,TPLANE,
                1QTIME0,QTIME1,DTOTIM,DT1,DTFR,RMININ,IQPOS
                2,STATS0,SAVADR,UNSADR,NUMSVR,IADDWS,ACTIM,
                3 NPRS,NLIN,IPLKPP,ITOTBK
      C COMMON BLOCK FOR THE .TF FILE VTTWS 370 WORDS
```

```
FORTRAN IV-PLUS V02-510          16:16:42     30-MAY-79
VTT.FTN          /TRIBLOCKS/WR

0016             COMMON /TIMR/ NQTS,NTPLNS,TIMVAL(5,25),RATIO(25),NSTEPS(25)
                 1,QTTME(5,28),QPTR(28)
        C COMMON BLOCK FOR THE ITTER PROGRAM
0017             COMMON /STATS/ITPNM,NSTPS,OMEG,CONVG,HIINT,IYHI,IXHI
                 1,IAHI,NITER,NSTP,IGX,TYMS,DTMS,HIDF(121)
        C COMMON BLOCK FOR THE VTT BOUNDARY LISTS FOR TIME VARING NODES
        C AND FOR SUBREGION B.C. GENERATION
        C        COMMON /LISTS/NOLRG,ISAVE(2,128),SPACE,NOHLD,IUNSAV(2,256),
        C        1SVAL(128),UNSVAL(256)
0018             COMMON /LISTS/NOHLD,IUNSAV(2,512),IEXTRA,UNSVAL(512)
        C
0019             EQUIVALENCE (FA(1),H2(1))
0020             EQUIVALENCE (BLKA(1),NUNK)
0021             EQUIVALENCE (RTIMES(1,1),IG(1))
0022             DATA ACOFF/ 1.,1.,1.,1.,0.,0.,0.,0.,0.,0.,0.,0.,.5,0.,.5,
                 11.,0.,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.
                 2,0.,1.,.5,0.,.5,1.,0.,0.,1.,0.,0.,.5,1.,0.,0.,1.,.5,0.,.5
                 3,1.,0.,0.,1.,.5,0.,.5,1.,0.,0.,1.,.5,0.,0.,1.,0.,0.,.5,.5,0.
                 4,0.,1.,1.,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.
                 5,1.,1.,0.,.5,1.,1.,0.,1.,1.,.5,.5,1.,1.,0./
0023             DATA AREA/1.,.5,.375,.625/
0024             DATA INDX3/1,2,0,2,  -1,1,-1,0,  -2,-1,0,-1/
        C OPEN FIELDS FILE, ANSWER FILE, AND QFILE AND READ IN DATA
0025                     WRITE(6,*) 'VTT TRANSIENT MODEL'
0026             WRITE(6,*) 'AC SWITCH 7 DOWN FOR ITTERATION PRINT'
0027             WRITE(6,*) 'AC SWITCH 7 UP FOR NO ITTERATION PRINT'
0028             WRITE(6,*) 'AC SWITCH 1 UP FOR RESTART'
0029             WRITE(6,*) 'AC SWITCH 1 DOWN FOR NEW RUN'
0030             WRITE(6,*) 'AC SWITCH 2 UP TO CHANGE OMEGA AND CONVERGENCE'
0031             WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0032             READ(6,*) IUN
0033             IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0034             TH0=SECNDS(0.)
0035             CALL OPFILE(2,'VTTFIELDS',2401.,IPTR)
0036             WRITE(6,1499)
0037     1499    FORMAT(' LARGE REGION OR SMALL (0 OR 1)')
0038             READ(6,1498) LRG
0039     1498    FORMAT(I5)
0040             IF(LRG.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,IPTR)
        C SET UP FILE OFFSETS
0041             IQOFF1=0
0042             IQOFF2=0
0043             IRSLTO=0
0044             IRSTOF=0
0045             ISTSOF=0
0046             IF(LRG.EQ.0) GO TO 1497
0047             IQOFF1=5601.
0048             IQOFF2=2.
0049             IRSLTO=20003.
0050             IRSTOF=1.
0051             ISTSOF=100.
0052     1497    CONTINUE
0053             CALL OPFILE(2,'VTTQS',11205.,IQPTR)
0054             CALL OPFILE(2,'VTTRESULT',40006.,IANS)
0055             CALL OPFILE(2,'VTTRESTRT',2.,IRSPTR)
0056             CALL OPFILE(2,'VTTSTATS',200.,ISTATS)
```

```
FORTRAN IV-PLUS V02-51D          16:16:42     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

0057             CALL DPFILE(2,'VTTLIST',624.,ILIST)
0058             CALL DPR(IPTR,0.,BLK0,100.)
0059             ITRDNT=1
0060             IF(DOIT1.NE.1) GO TO 30138
0061             WRITE(6,*) 'TRIDENT TYPE AQUIFER'
0062             WRITE(6,*) 'ENTER 0 FOR NEW LEAKY AQUIFER'
0063             WRITE(6,*) 'ENTER 1 FOR NO DEEP AQUIFER'
0064             WRITE(6,*) 'ENTER 2 FOR OLD LEAKY TYPE AQUIFER'
0065             READ(6,1498) ITRDNT
0066     30138   WRITE(6,*) 'IS THERE AN ADDITIONAL DAILY Q-FILE CR=NO 1=YES'
0067             READ(6,1498) IADDQS
0068             IF(IADDQS.EQ.1) CALL DPFILE(2,'VTTADDQS',367.,IADQ)
0069             IF(IADDQS.EQ.1) CALL DPR(IADQ,1.,IQTIM0(1),248.)
0070             IF(IADDQS.EQ.1) NWLLS=NWLLSL
0071             IF(IADDQS.EQ.1.AND.LRG.EQ.1) NWLLS=NWLLSS
         C CODING REQUIRED FOR COMPRESSION OF TIME STEP AND TIME PLANE
         C POTENTIAL DATA IN THE RESULT FILE AND THE CODING TO RETRIVE
         C THIS COMPRESSED DATA AND PUT INTO THE VTTFIELDS FILE
         C
         C THIS FOLLOWING PROCEEDURE NEEDS ONLY BE DONE ONCE FOR AND GIVEN
         C SYSTEM IT CALCULATES SPACE REQUIRED AND THE NUMBER OF PAGES OF
         C DATA NEEDED FOR THE COMPRESSION
0072                     NLIN=(4608-128)/NCOLS
0073                     IF(NLIN.GT.NLINES) NLIN=NLINES
0074                     NPGS=NLINES/NLIN
0075                     IF(MOD(NLINES,NLIN).NE.0) NPGS=NPGS+1
         C NPG=TOTAL NUMBER OF PAGES REQUIRED
         C NLIN=NUMBER OF MATRIX LINES PER PAGE NOT COUNTING THE 1 LINE OVERLAP
         C NCOLS=TOTAL NUMBER OF COLUMNS PER MATRIX LINE
         C NLINES=TOTAL NUMBER OF MATRIX LINES
         C TOTAL SIZE OF BUFFER REQUIRED FOR THIS PROCEEDURE IS 4608 REAL WORDS
         C TO ALLOW FOR THE ONE LINE OVERLAP
         C NOW CALCULATE THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR STORAGE
         C OF A TIME PLANE
         C
         C ADJUST THE LINE COUNT FOR OVERLAP
0076                     IF(NLIN.NE.NLINES) NLINA=NLIN+1
         C CALCULATE THE DISK BLOCKS REQUIRED FOR ONE FULL PAGE
0077                     IBLKPP=NLINA*NCOLS/128
0078                     IF(MOD(NLINA*NCOLS,128).NE.0) IBLKPP=IBLKPP+1
0079                     ITOTBK=IBLKPP*(NPGS-1)
         C ITOTBK IS THE BLOCKS REQUIRED FOR FULL PAGES NOW CALCULATE
         C THE BLOCKS REQUIRED FOR THE LAST PAGE
0080                     LINLFT=NLINES-NLIN*(NPGS-1)
0081                     IBLKLF=LINLFT*NCOLS/128
0082                     IF(MOD(LINLFT*NCOLS,128).NE.0) IBLKLF=IBLKLF+1
0083                     ITOTBK=ITOTBK+IBLKLF
         C NOW ITOTBK IS THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR TIME PLANE
         C SAVING.
         C
         C CHECK FOR INCONSISTENCY
0084             IF(LRG.NE.ILRGSM) STOP
0085             SAVADR=16.
0086             UNSADR=216.
         C       NOLRG=0
         C       IF(ISHLSM.EQ.1) CALL DPR(ILIST,8.,NOLRG,258.)
```

```
0087              NOHLD=0
0088              IF(ITVN.NE.1) GO TO 1493
0089              CALL DPR(ILIST,11.,NOHLD,1026.)
0090      1493    CONTINUE
0091              CALL DPR(IQPTR,11201.+IQOFF2,NQTS,370.)
0092              CALL DPR(IANS,0.+IRSLTO,NUMSAV,602.)
0093              STOFCT=STOMUL
0094              NWDS=NCOLS*2.
0095              ISWV=2
0096              CALL SSWTCH(1,ISWV)
0097              IF(ISWV.NE.1) GO TO 1500
          C RESTART CASE
0098              CALL DPR(IRSPTR,0.+IRSTOF,LTMPLN,60.)
0099              WRITE(6,1501) (TITLE(I),I=1,71),RMININ,LTMPLN
0100      1501    FORMAT(' VTT RESTART CASE FOR'/1X,71A1/
                  1' CURRENTLY',F12.2,' MINUTES INTO THE SOLUTION'/
                  1' NEXT TIME PLANE TO BE SAVED WILL BE',I5)
          C       DO 1502 I=1,NLINES
          C       ADR0=(LTMPLN-1)*200.+I+3.-1.
          C       ADR1=(I-1)*12.+1.
          C       ADR2=ADR1+9.
          C       CALL DPR(IANS,ADR0+IRSLTO,H0,NWDS)
          C       CALL DPW(IPTR,ADR1,H0,NWDS)
          C       CALL DPW(IPTR,ADR2,H0,NWDS)
          C1502   CONTINUE
          C THIS PROCEDURE READS THE RESULT FILE AND RESTORES THE VTTFIELDS FILE
          C FROM THE COMPRESSED FASHION ALLOWING A 1 LINE OVERLAP PER PAGE
          C OFFSET IS IRSLTO+3.
          C NUMSAV IS THE TIME STEP NUMBER
0101                    DO 19133 IPG=1,NPGS
0102                    ISTLN=(IPG-1)*NLIN+1
0103                    ISTPLN=ISTLN-1+NLIN
0104                    IF(ISTPLN.GT.NLINES) ISTPLN=NLINES
0105                    IF(IPG.NE.NPGS) ISTPLN=ISTPLN+1
0106                    WDS=NCOLS*2.
0107                    WRDS=WDS
0108                    BUFWDS=(ISTPLN-ISTLN+1)*NCOLS*2.
0109                    ADR=IRSLTO+3.+ITOTBK*(NUMSVR-1)+(IPG-1)*IBLKPP
0110                    CALL DPR(IANS,ADR,FA,BUFWDS)
0111                    DO 19132 IL=ISTLN,ISTPLN
0112                    ADR=(IL-1)*12.+1.
0113                    IPOS=(IL-ISTLN)*NCOLS+1
0114                    CALL DPW(IPTR,ADR,FA(IPOS),WRDS)
0115                    CALL DPW(IPTR,ADR+9.,FA(IPOS),WRDS)
0116      19132         CONTINUE
0117      19133         CONTINUE
          C
          C
0118              CALL DPR(IANS,0.+IRSLTO,NUMSAV,602.)
0119              NUMSAV=NUMSVR
0120              ISTPLN=LTMPLN
0121              IF(IADDQS.EQ.1) CALL DPFILE(2,'VTTADDQS',367.,IADQ)
0122              IF(IADDQS.EQ.1) CALL DPR(IADQ,1.,IQTIM0(1),248.)
0123              IF(IADDQS.EQ.1) NWLLS=NWLLSL
0124              IF(IADDQS.EQ.1.AND.LRG.EQ.1) NWLLS=NWLLSS
0125              GO TO 25
```

```
FORTRAN IV-PLUS V02-51D          16:16:42     30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0126     1500     DO 15 I=1,512
0127     15       IG(I)=0.
         C ZERO THE STAT FILE
0128              DO 155 STA=0.,99.,1.
0129              CALL DPW(ISTATS,STA+ISTSOF,IG,256.)
0130     155      CONTINUE
         C        NUMSAV=1
         C        DO 16 I=1,NLINES
         C        ADR=(I-1)*12.+1.
         C        CALL DPR(IPTR,ADR,H0,NWDS)
         C        ADR1=I+3.-1.
         C        CALL DPW(IANS,ADR1+IRSLTO,H0,NWDS)
         C        CALL DPW(IPTR,ADR+9.,H0,NWDS)
         C16      CONTINUE
         C THIS PROCEEDURE READS THE VTTFIELDS FILE AND SAVES TO THE RESULT FILE
         C IN COMPRESSED FASHION ALLOWING A 1 LINE OVERLAP PER PAGE
         C OFFSET IS IRSLTO+3.
         C NUMSAV IS THE TIME STEP NUMBER
0131                       NUMSAV=0
0132                       NTPLSV=0
0133                       DO 30133 IPG=1,NPGS
0134                       ISTLN=(IPG-1)*NLIN+1
0135                       ISTPLN=ISTLN-1+NLIN
0136                       IF(ISTPLN.GT.NLINES) ISTPLN=NLINES
0137                       IF(IPG.NE.NPGS) ISTPLN=ISTPLN+1
0138                       WDS=NCOLS*2.
0139                       WRDS=WDS
0140                       DO 30132 IL=ISTLN,ISTPLN
0141                       ADR=(IL-1)*12.+1.
0142                       IPOS=(IL-ISTLN)*NCOLS+1
0143                       CALL DPR(IPTR,ADR,FA(IPOS),WRDS)
0144                       CALL DPW(IPTR,ADR+9.,FA(IPOS),WRDS)
0145     30132             CONTINUE
0146                       BUFWDS=(ISTPLN-ISTLN+1)*NCOLS*2.
0147                       ADR=IRSLTO+3.+ITOTBK*NUMSAV+(IPG-1)*IBLKPP
0148                       CALL DPW(IANS,ADR,FA,BUFWDS)
0149     30133             CONTINUE
0150                       NUMSAV=NUMSAV+1
0151                       NTPLSV=NTPLSV+1
0152                       ISAVNO(NTPLSV)=NUMSAV-1
         C
         C
0153              RTIMES(1,1)=YEAR
0154              RTIMES(2,1)=MONTH*100+DAY
0155              RTIMES(3,1)=HOUR*100+MIN
0156              RMIN(1)=0.
0157              CALL DPW(IANS,0.+IRSLTO,NUMSAV,602.)
0158              IQPOS=2
         C CALCULATE A BASE TIME FOR JAN 1 HOUR 0 MIN 0
0159                       ITIMC(1)=YEAR
0160                       ITIMC(2)=1
0161                       ITIMC(3)=0
0162                       ITIMC(4)=0
0163                       ITIMC(5)=0
0164                       CALL TIMHRS(ITIMC,TT1,TTFR)
0165                       DT1=TT1
```

```
0166                    DTFR=TTFR
0167                    ACTIM=DT1+DTFR
      C                                          WRITE(6,*) 'ACTIM',ACTIM
0168            CALL TIMHRS(YEAR,TT1,TFR)
0169            DT1=TT1
0170            DTFR=TFR
0171            STIME=DT1+DTFR
0172            TIME=STIME
0173            CALL TIMHRS(YEARL,TT1,TFR)
0174            DT1=TT1
0175            DTFR=TFR
0176            ETIME=DT1+DTFR
0177            STIMEE=0.0
0178            ETIMEE=ETIME-STIME
0179            TIMEE=TIME-STIME
0180            CALL TIMHRS(QTIME(1,1),TT1,TFR)
0181            DT1=TT1
0182            DTFR=TFR
0183            QTIME2=DT1+DTFR
0184            CALL TIMHRS(QTIME(1,2),TT1,TFR)
0185            DT1=TT1
0186            DTFR=TFR
0187            QTIME1=DT1+DTFR
0188            ISTPLN=1
      D         WRITE(5,1505)
      D         1 NUNK,IHBW,NLTNES,NCOLS,NITT,ITTCNT,CONV,
      D         1BIAS,NADS,DXFER,IROT,MAXIT,DELX,AUTO,OMEGA,NTMPLN,
      D         2NOTIHS,TRNOSS,IWID,TIME,STIME,ETIME,YEAR,MONTH,DAY,HOUR,
      D         3MIN,YEARL,MONTHL,DAYL,HOURL,MINL,NOSX,NOSY,NUCONF,
      D         4DDIT,DDIT1
      D1505     FORMAT(6I10/2F17.2,I10,I2,I2,I10,F17.2,I2,F5.2/
      D         10I10/3E17.7/10I5,3I10,A1)
      D         WRITE(5,1606) STIME,ETIME,QTIME0,QTIME1
      D         CLOSE(UNIT=5)
      D1606     FORMAT(' TIMES'/4E19.8)
0189            STATSM=0.
0190            IF(AUTO.EQ.1) GO TO 25
      C MANUAL CONTROL INPUT INTERACTION
      C READY TO START
0191  25        CONTINUE
      C ACCOUNT FOR THE ADDITIONAL Q'S IF ANY
0192                    IF(IADDQS.NE.1) GO TO 25833
      C CALCULATE THE NUMBER OF DAYS FROM THE STARTING DAY
0193                    DT1=TIME-ACTIM+1.0E-10
      C                                 WRITE(6,*) 'T,A,D',TIME,ACTIM,DT1
0194                    IDAYQL=DT1
      C                                 WRITE(6,*) 'IDAYQL',IDAYQL
0195                    ADRQ=IDAYQL
0196                    CALL DPR(IADQ,ADRQ,IQTIM0(1),248,)
0197                    NWLLS=NWLLSL
0198                    IF(LRG.EQ.1) NWLLS=NWLLSS
0199                    IPUSQ=1
0200  25833             CONTINUE
0201            DO 5710 ILN=1,NLINES
0202            QADDR=ILN+IQOFF1+200.*(QPTR(IQPOS-1)-1)
0203            QADDR1=QADDR+210.
```

```
0204            CALL DPR(IQPTR,QADDR,Q0,NWDS)
0205            CALL DPR(IQPTR,QADDR1,QNEX0,NWDS)
0206                    IF(IADDQS.NE.1) GO TO 25832
0207    25830           IF(NWLLS.LE.0) GO TO 25832
0208                    IF(IPOSQ.GT.NWLLS) GO TO 25832
0209                    LINQ=IWLOCL(2,IPOSQ)
0210                    ICLQ=IWLOCL(1,IPOSQ)
0211                    IF(LRG.EQ.1) LINQ=IWLOCS(2,IPOSQ)
0212                    IF(LRG.EQ.1) ICLQ=IWLOCS(1,IPOSQ)
0213                    IF(LINQ.NE.ILN) GO TO 25832
0214                    QRAT=WLRATL(IPOSQ)
0215                    IF(LRG.EQ.1) QRAT=WLRATS(IPOSQ)
0216                    Q0(ICLQ)=Q0(ICLQ)+QRAT
        C                       WRITE(6,*) 'IPOSQ',IPOSQ
        C                       WRITE(6,*) 'LINQ',LINQ
        C                       WRITE(6,*) 'ILN,ICLQ,QRAT',ILN,ICLQ,QRAT
0217                    QNEX0(ICLQ)=QNEX0(ICLQ)+QRAT
0218                    IPOSQ=IPOSQ+1
0219                    GO TO 25830
0220    25832           CONTINUE
0221            QADR=(ILN-1.)*12.+4.
0222            QADR1=(ILN-1.)*12.+12.
0223            CALL DPW(IPTR,QADR,Q0,NWDS)
0224            CALL DPW(IPTR,QADR1,QNEX0,NWDS)
0225    5710    CONTINUE
0226            DO 70 NTPLN=ISTPLN,NTMPLN
0227            ITPNM=NTPLN
0228            LTMPLN=NTPLN+1
0229            NT=NSTEPS(NTPLN)
0230            RAT=1./RATIO(NTPLN)
0231            CALL TIMHRS(TIMVAL(1,NTPLN),TT1,TFR)
0232            DT1=TT1
0233            DTFR=TFR
0234            TPLANE=DT1+DTFR
0235            DTOTIM=DT1+DTFR-TIME
0236            TOTTIM=DTOTIM
0237            ATIM=TOTTIM/NT
0238            IF(RAT.EQ.1.) GO TO 26
0239            ATIM=TOTTIM*((1.-RAT)/(1.-RAT**NT))
0240    26      CONTINUE
        D       WRITE(5,2606) NTPLN,NT,NSTEPS(NTPLN),RATIO(NTPLN),
        D       1RAT,TPLANE,TIME,TOTTIM,ATIM
        D2606   FORMAT(' DO 70 DO LOOP'/3I10/2E20.7/2E20.7/2E20.7)
        D       CLOSE(UNIT=5)
0241            NSTPS=NSTEPS(NTPLN)
0242            DO 60 NSTEP=1,NSTEPS(NTPLN)
        C UPDATE THE HELD NODES IF NECESSARY
0243            IF(ITVN.NE.1) GO TO 1489
0244            CALL DPR(ILIST,UNSADR,UNSVAL,1024.)
        C       WRITE(5,9076) UNSADR
        C9076   FORMAT(F10.2)
        C       WRITE(5,9078) NOHLD
        C9078   FORMAT(10(2I4,2X))
        C       WRITE(5,9078) IUNSAV
        C       WRITE(5,9077) UNSVAL
        C9077   FORMAT(10F10.2)
```

```
FORTRAN IV-PLUS V02-510          16:16:42    30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0245             NOD=1
0246             IF(NOD.GT.NOHLD) GO TO 1489
0247             DO 1492 ILINX=1,NLINES
0248             IF(IUNSAV(2,NOD).EQ.0) GO TO 1490
0249             IF(IUNSAV(2,NOD).NE.ILINX) GO TO 1492
0250     1490    ADRX=(ILINX-1)*12.+1.
0251             CALL DPR(IPTR,ADRX,H0,NWDS)
         C       WRITE(5,9074) ILINX,ADRX
         C9074   FORMAT(' LINE=',I5,F10.1)
         C       WRITE(5,9075) (H0(IU),IU=1,NCOLS)
         C9075   FORMAT(10F10.2)
0252             IF(IUNSAV(2,NOD).EQ.0) GO TO 14915
0253     1491    H0(IUNSAV(1,NOD))=UNSVAL(NOD)
0254     14915   NOD=NOD+1
0255             IF(NOD.GT.NOHLD) GO TO 1488
0256             IF(IUNSAV(2,NOD).EQ.0) GO TO 14915
0257             IF(IUNSAV(2,NOD).EQ.ILINX) GO TO 1491
0258     1488    CONTINUE
0259             CALL DPW(IPTR,ADRX,H0,NWDS)
         C       WRITE(5,9075) (H0(IU),IU=1,NCOLS)
0260             IF(NOD.GT.NOHLD) GO TO 1489
0261     1492    CONTINUE
0262     1489    CONTINUE
0263             TM00=SECNDS(0)
0264             STATS=STATS0+NSTEP
0265             NSTP=NSTEP
0266             OMEG=OMEGA
0267             CONVG=CONV
0268             HIINT=0.
0269             NITER=0
0270             DELTIM=ATIM*RAT**(NT-NSTEP)
0271             TIME0=TIME
0272             TIME=TIME+DELTIM
0273             RMININ=(TIME-STIME)*24.*60.
0274                     DT1=TIME-ACTIM+1.E-10
         C                   WRITE(6,*) 'T,A,D',TIME,ACTIM,DT1
0275                     IDAYQ=DT1
         C                   WRITE(6,*) 'IDAYQ',IDAYQ
0276                     IF(IADDGS.NE.1) GO TO 25820
0277                     IF(NWLLS.EQ.0) GO TO 25820
0278                     IF(IDAYQ.EQ.IDAYQL) GO TO 25820
         C WE HAVE GONE TO A NEW DAY SO UPDATE DAILY Q'S
0279                     ADRQ=IDAYQ
0280                     CALL DPR(IADQ,ADRQ,IQTIM0(1),248.)
0281                     NWLLS=NWLLSL
0282                     IF(LRG.EQ.1) NWLLS=NWLLSS
0283                     IDAYQL=IDAYQ
0284                     IPOSQ=1
0285                     LINQL=IWLOCL(2,IPOSQ)
0286                     IF(LRG.EQ.1) LINQL=IWLOCS(2,IPOSQ)
0287                     QADDR=LINQL+IQOFF1+200.*(QPTR(IQPOS-1)-1)
0288                     QADDR1=QADDR+200.
0289                     QADR=(LINQL-1)*12.+4.
0290                     QADR1=(LINQL-1)*12.+12.
0291                     CALL DPR(IQPTR,QADDR,Q0,NWDS)
0292                     CALL DPR(IQPTR,QADDR1,QNEX0,NWDS)
```

```
0293    25818           LINQ=IWLOCL(2,IPOSQ)
0294                    ICLQ=IWLOCL(1,IPOSQ)
0295                    IF(LRG.EQ.1) LINQ=IWLOCS(2,IPOSQ)
0296                    IF(LRG.EQ.1) ICLQ=IWLOCS(1,IPOSQ)
0297                    IF(IPOSQ.GT.NWLLS) GO TO 25819
0298                    IF(LINQ.NE.LINQL) GO TO 25819
0299    25817           QRAT=WLRATL(IPOSQ)
0300                    IF(LRG.EQ.1) QRAT=WLRATS(IPOSQ)
0301                    Q0(ICLQ)=Q0(ICLQ)+QRAT
        C                       WRITE(6,*) 'IPOSQ',IPOSQ
        C                       WRITE(6,*) 'LINQ,ICLQ,QRAT',LINQ,ICLQ,QRAT
0302                    QNEX0(ICLQ)=QNEX0(ICLQ)+QRAT
0303                    IPOSQ=IPOSQ+1
0304                    GO TO 25818
0305    25819           CALL DPW(IPTR,QADR,Q0,NWDS)
0306                    CALL DPW(IPTR,QADR1,QNEX0,NWDS)
0307                    IF(IPOSQ.GT.NWLLS) GO TO 25820
0308                    LINQL=LINQ
0309                    QADDR=LINQL+IQOFF1+200.*(QPTR(IQPOS-1)-1)
0310                    QADDR1=QADDR+200.
0311                    QADR=(LINQL-1)*12.+4.
0312                    QADR1=(LINQL-1)*12.+12.
0313                    CALL DPR(IQPTR,QADDR,Q0,NWDS)
0314                    CALL DPR(IQPTR,QADDR1,QNEX0,NWDS)
0315                    GO TO 25817
0316    25820           CONTINUE
0317    2610    CONTINUE
        D       WRITE(5,2616) NSTEP,NT,RAT,ATIM,TIME0,TIME,DELTIM
        02616   FORMAT(' DO 60 DO LOOP'/2I10/2E20.7/3E20.7,2X,' CONTINUE')
        D       CLOSE(UNIT=5)
        D       READ(6,2706) IGHJ
        D       WRITE(5,2773) IGHJ
        02773   FORMAT(' IGHJ=',I5)
        02706   FORMAT(I5)
0318            IF(TIME.GE.QTIME0.AND.TIME.LE.QTIME1) GO TO 28
        C MUST ADVANCE Q'S
0319    27      IQPOS=IQPOS+1
        C       WRITE(5,2806) IQPOS
        D       CLOSE(UNIT=5)
        C2806   FORMAT(' ADVANCE QS',1X,I5)
0320            IQWUPTR(IQPOS)
0321            QTIME0=QTIME1
        C       WRITE(5,7689) (QTIME(L,IQPOS),L=1,5)
        C7689   FORMAT(' YR,MON,DAY,HR,MIN'/ 5I10)
0322            CALL TIMHRS(QTIME(1,IQPOS),TT1,TFR)
0323            DT1=TT1
0324            DTFR=TFR
0325            QTIME1=DT1+DTFR
        C       WRITE(5,5678) QTIME0,QTIME1,IQPOS,IQ
        C5678   FORMAT(' QTIME0,QTIME,IQPOS,IQ'/2E20.6,2I10)
0326                    IPOSQ=1
0327            DO 2710 ILN=1,NLINES
0328            QADDR=200.*(IQ-1)+ILN
0329            CALL DPR(IQPTR,QADDR+IQOFF1,QNEX0,NWDS)
0330                    IF(IADDQS.NE.1) GO TO 26832
0331    26830           IF(NWLLS.LE.0) GO TO 26832
```

```
0332                    IF(IPOSQ.GT.NWLLS) GO TO 26832
0333                    LINQ=IWLOCL(2,IPOSQ)
0334                    ICLQ=IWLOCL(1,IPOSQ)
0335                    IF(LRG.EQ.1) LINQ=IWLOCS(2,IPOSQ)
0336                    IF(LRG.EQ.1) ICLQ=IWLOCS(1,IPOSQ)
0337                    IF(LINQ.NE.ILN) GO TO 26832
0338                    QRAT=WLRATL(IPOSQ)
0339                    IF(LRG.EQ.1) QRAT=WLRATS(IPOSQ)
0340                    QNEX0(ICLQ)=QNEX0(ICLQ)+QRAT
        C                       WRITE(6,*) 'IPOSQ',IPOSQ
        C                       WRITE(6,*) 'ILN,LINQ,ICLQ',ILN,LINQ,ICLQ
0341                    IPOSQ=IPOSQ+1
0342                    GO TO 26830
0343    26832           CONTINUE
0344            QADDRN=(ILN-1)*12.+12.
0345            QADDR0=(ILN-1)*12.+4.
0346            CALL DPR(IPTR,QADDRN,Q0,NWDS)
0347            CALL DPW(IPTR,QADDRN,QNEX0,NWDS)
0348            CALL DPW(IPTR,QADDR0,Q0,NWDS)
0349    2710    CONTINUE
0350            GO TO 2610
0351    28      CONTINUE
0352            IF(TIME.GT.QTIME1) GO TO 27
0353            QTIM=QTIME0
0354            QTIMP=QTIME1
0355            ATIME=TIME
0356            DTIME=DELTIM
0357            OCEAN=SEALEV
0358            ITTCNT=0
0359    29      IHIAQ=0
0360            HIDF(ITTCNT+1)=0
0361            IHICOL=0
0362            IHILIN=0
0363            HIDIF=0.0
        C LOOP TO CALCULATE THE VALUES FOR ONE ITTERATION
0364            DO 50 ILINE=1,NLINES
        D       IF(ILINE.NE.IGHJ) GO TO 8976
        D       WRITE(5,2006)
        D2006   FORMAT(' DO LOOP 50 ')
        C SHIFT LINES 4 AND 0 TO 0 AND 2
        D       IF(ILINE.EQ.7) WRITE(5,2108) ACOFF
        D2108   FORMAT(4F19.5)
        D8976   CONTINUE
0365            CALL SHIFT(FA,NCOLS)
        C READ IN LINE ILINE+1
0366            BLK=(ILINE)*12.+1.
0367            IF (ILINE.NE.NLINES)CALL DPR(IPTR,BLK,FA(1),3072.)
0368            IF(ILINE.EQ.1) CALL DPR(IPTR,1.,FA(1537),3072.)
        D       IF(ILINE.NE.IGHJ) GO TO 8977
        D       WRITE(5,2107) ILINE
        D2107   FORMAT(/' LINE='I5)
        D       DO 8999 I=1,NCOLS
        D       WRITE(5,2105) (I,H4(I),B4(I),K4(I),Q4(I),NTYP4(I),
        D       1NPOS4(I),TOP4(I),COEF4(I),OCOEF4(I),OH4(I),STO4(I),
        D       1QNEX4(I),
        D       II,H0(I),B0(I),K0(I),Q0(I),NTYP0(I),NPOS0(I),TOP0(I),
```

```
FORTRAN IV-PLUS V02-51D          16:16:42     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

        D        1COEF0(I),OCOEF0(I),OH0(I),STO0(I),QNEX0(I),
        D        1I,H2(I),B2(I),K2(I),Q2(I),NTYP2(I),NPOS2(I),TOP2(I),
        D        1COEF2(I),OCOEF2(I),OH2(I),STO2(I),QNEX2(I))
        D2105    FORMAT(I5/4E19.8,2I5,E20.8/5E20.8)
        D8999    CONTINUE
        D        CLOSE(UNIT=5)
        D8977    CONTINUE
0369             IF(NPOS0(1).EQ.0) GO TO 46
0370             DO 40 INODE=1,NPOS0(1)
        D        IF(ILINE.NE.IGHJ) GO TO 9877
        D        WRITE(5,2106) NPOS0(INODE+1)
        D2106    FORMAT(' DO 40 DO LOOP',1X,I5)
        D9877    CONTINUE
0371             IAQ=1
0372             IPOS=NPOS0(INODE+1)
0373             IF(IPOS.GT.IWID) IAQ=2
0374             IF(IPOS.GT.IWID*2) IAQ=3
0375             H(1)=OH0(IPOS)
0376             HP(1)=H0(IPOS)
0377             K(1)=K0(IPOS)
0378             BOT(1)=B0(IPOS)
0379             TOP(1)=TOP0(IPOS)
0380             S=STO0(IPOS)
0381             CO=OCOEF0(IPOS)
0382                     IF(IAQ.NE.1.AND.DOIT1.EQ.1) CO=0.
0383             Q=Q0(IPOS)
0384             QP=QNEX0(IPOS)
0385             H(2)=OHV(IPOS+1)
0386             HP(2)=H0(IPOS+1)
0387             K(2)=K0(IPOS+1)
0388             BOT(2)=B0(IPOS+1)
0389             TOP(2)=TOP0(IPOS+1)
0390             H(3)=OH2(IPOS)
0391             HP(3)=H2(IPOS)
0392             K(3)=K2(IPOS)
0393             BOT(3)=B2(IPOS)
0394             TOP(3)=TOP2(IPOS)
0395             H(4)=OH0(IPOS-1)
0396             HP(4)=H0(IPOS-1)
0397             K(4)=K0(IPOS-1)
0398             BOT(4)=B0(IPOS-1)
0399             TOP(4)=TOP0(IPOS-1)
0400             H(5)=OH4(IPOS)
0401             HP(5)=H4(IPOS)
0402             K(5)=K4(IPOS)
0403             BOT(5)=B4(IPOS)
0404             TOP(5)=TOP4(IPOS)
0405                     C6=0.
0406                     C7=0.
0407                     H(6)=0.
0408                     H(7)=0.
0409                     POTDIF=0.
0410                     AREF=1.
0411                     ARF=0.
0412                     HP(6)=0.
0413                     HP(7)=0.
```

```
0414                    NT6=1
0415                    NT7=1
0416                    IH6=INDX3(1,IAQ)*IWID+IPOS
0417                    IH7=INDX3(2,IAQ)*IWID+IPOS
0418                    IT6=INDX3(3,IAQ)*IWID+IPOS
0419                    IT7=INDX3(4,IAQ)*IWID+IPOS
0420            IF(DOIT1.NE.1) GO TO 195
        C IF NO LEAKAGE GO TO 35
0421                    IF(ITRDNT.EQ.1) GO TO 35
        C IF NEW TRIDENT TYPE LEAKAGE GO TO 1951
0422                    IF(ITRDNT.EQ.0) GO TO 1951
        C OLD TRIDENT LEAKAGE
        C NOW IF WE ARE IN AQUIFER 2 THEN NO LEAKAGE IS CALCULATED
0423                    IF(IAQ.NE.1) GO TO 35
        C WE ARE IN AQUIFER 1 WITH OLD TYPE TRIDENT LEAKAGE
0424            C6=COFLEK
0425            H(6)=COEF0(IPOS)
0426            HP(6)=H(6)
0427            C7=0.
0428            H(7)=0.
0429            HP(7)=0.
0430                    NT6=NTYP0(2*IH6)
0431                    ARFF=AREA(NT6/8+1)
0432                    IF(NT6.EQ.1) ARFF=0.
0433            GO TO 35
        C NEW TRIDENT TYPE LEAKAGE
0434    1951            H(6)=OH0(IH6)
0435                    HP(6)=H0(IH6)
0436                    NT6=NTYP0(2*IH6)
0437                    ARFF=AREA(NT6/8+1)
0438                    IF(NT6.EQ.1) ARFF=0.
0439                    C6=COFLEK
0440                    POTDIF=2.*(COEF0(IPOS)-COEF0(IH6))
0441                    GO TO 35
0442    195     CONTINUE
        C       C6=0.
        C       C7=0.
        C       H(6)=0.
        C       H(7)=0.
        C       HP(6)=0.
        C       HP(7)=0.
        C       NT6=1
        C       NT7=1
        C       IH6=INDX3(1,IAQ)*IWID+IPOS
        C       IH7=INDX3(2,IAQ)*IWID+IPOS
        C       IT6=INDX3(3,IAQ)*IWID+IPOS
        C       IT7=INDX3(4,IAQ)*IWID+IPOS
0443            GO TO (35,355,3555),NAQFRS
0444    3555    H(7)=OH0(IH7)
0445            HP(7)=H0(IH7)
0446            NT7=NTYP0(2*IH7)
0447            IF(NT7.NE.1) C7=COEF0(IT7)
0448    355     H(6)=OH0(IH6)
0449            HP(6)=H0(IH6)
0450            NT6=NTYP0(2*IH6)
0451            IF(NT6.NE.1) C6=COEF0(IT6)
```

```
FORTRAN IV-PLUS V02-51D          16:16:42     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

        C NOW SET UP NODE TYPE ARRAY
0452    35      CONTINUE
0453            NDTYPA=NTYP0(2*(IPOS-1))
0454            NDTYPC=NTYP0(2*(IPOS+1))
0455            NDTYP=NTYP0(2*IPOS)
0456            ARA=AREA(NDTYP/8+1)
0457            DLX2=DELX*DELX*ARA
0458            IF(NDTYP.GT.2) NDTYP=NDTYP-5
0459            NDTYP=NDTYP+1
        C       IF(DOIT1.EQ.1) C6=C6*ARA
0460                    IF(DOIT1.EQ.1.AND.ITRDNT.NE.1) C6=C6*AMIN1(ARFF,ARA)
        C               IF(DOIT1.EQ.1.AND.ITRDNT.NE.1) ARF=AMIN1(ARF,ARFF)
0461            DO 36 I=1,4
0462    36      A0(I)=ACOFF(I,NDTYP)
        D       IF(ILINE.NE.IGHJ) GO TO 9873
        D       WRITE(5,3505) NDTYP,A0,DLX2
        03505   FORMAT(I5,5F20.5)
        D       WRITE(5,3606) IPOS
        03606   FORMAT(' LOCAL VARIABLES IPOS=',I5)
        D       WRITE(5,3605) H,HP,T,TP,K,BOT,TOP,S,C6,C7,CD,A0,Q,QP,
        D       1QTIM,QTIMP,ATIME,DTIME,DLX2,T0,OCEAN
        03605   FORMAT(2(7E16.8/),5(5E19.8/),4E20.8/4E20.8/5E20.8/2E20.8/
        D       1QE20.8/E20.8)
        D       CLOSE(UNIT=5)
        09873   CONTINUE
0463            CALL LOCAL(INODE,IGHJ,ILINE,NDTYPA,NDTYPC)
0464    40      CONTINUE
0465            CALL TRIDIG(1,NPOS0(1))
        C CHECK H1 DIFFERENCE
        D       IF(ILINE.NE.IGHJ) GO TO 9872
        D       WRITE(5,4005) (Z(I),I=1,NPOS0(1))
        04005   FORMAT(5E20.8/)
        D       WRITE(5,4005) (H0(I),I=1,NCOLS)
        09872   CONTINUE
0466            DO 42 INODE=1,NPOS0(1)
0467            IPOS=NPOS0(INODE+1)
0468            IF(B0(IPOS).LT.0) GO TO 410
0469            IF (Z(INODE)-B0(IPOS).LE.0)Z(INODE)=B0(IPOS)+.001
0470    410     DIF=Z(INODE)-H0(IPOS)
0471            IF(ABS(DIF).LT.ABS(HIDIF)) GO TO 41
0472            HIDIF=DIF
0473            IHILIN=ILINE
0474            IHIAQ=1+(IPOS-1)/IWID
0475            IHICOL=IPOS-((IPOS-1)/IWID)*IWID
0476    41      CONTINUE
0477            H0(IPOS)=H0(IPOS)+OMEGA*DIF
0478    42      CONTINUE
0479            BLK=(ILINE-1)*12.+1.
0480            CALL OPAC(IPTR,BLK,H0,NWDS)
0481    46      CONTINUE
0482    50      CONTINUE
0483            ITTCNT=ITTCNT+1
0484            HIDF(ITTCNT)=HIDIF
0485            IF(ABS(HIDIF).LT.ABS(HIINT)) GO TO 501
0486            HIINT=HIDIF
0487            IYHI=IHILIN
```

```
FORTRAN IV-PLUS V02-510          16:16:42     30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0488                IXHI=IHICOL
0489                IAHI=IHIAQ
0490      501       CONTINUE
0491                ISWV=2
0492                CALL SSWTCH(7,ISWV)
0493                IF(ISWV.EQ.1) GO TO 90087
0494                WRITE(6,650) RMININ,NSTEP,NSTEPS(NTPLN),NTPLN,ITTCNT,
                    1IHILIN,IHICOL,IHIAQ,HIDIF
0495      90087     CONTINUE
0496      650       FORMAT(' MINUTES INTO SIMULATION=',F12.2/
                    1' STEP ',I5,' OF ',I5,' STEPS FOR TIME PLANE',I5/
                    1' ITTERATION=',I4/' LINE=',I4,' COL.=',I4,' AQUIFER=',I4
                    1/' DIFFERENCE=',F9.5/)
0497                IF(ITTCNT.GE.MAXIT) GO TO 59
0498                IF(ABS(HIDIF).LT.ABS(CONV)) GO TO 59
          C MANUAL INTERACTION
0499                ISWV=2
0500                CALL SSWTCH(2,ISWV)
0501                IF(ISWV.NE.1) GO TO 29
          C CHANGE VALUES
0502                WRITE(6,6006) OMEGA,CONV
0503                READ(6,7007) ICHG
0504                IF(ICHG.EQ.0) GO TO 29
0505                WRITE(6,7008)
0506                READ(6,7009) OMEGA,CONV
0507                WRITE(6,7009) OMEGA,CONV
0508                GO TO 29
0509      59        CONTINUE
0510                TTMS=SECNDS(TM0)
0511                DTMS=SECNDS(TM00)
0512                NITER=ITTCNT
          C SAVE LARGE GRID VALUES AND UPDATE SAVADR & UNSADR
          C         IF(ISMLSM.EQ.0) GO TO 1389
          C         NOD=1
          C         DO 1388 IKK=1,NOLRG
          C         SVAL(IKK)=0.
          C1388     CONTINUE
          C         IF(NOD.GT.NOLRG) GO TO 1390
          C         DO 1392 ILINX=1,NLINES
          C         IF(ISAVE(2,NOD).NE.ILINX) GO TO 1392
          C         ADRX=(ILINX-1)*12.+1.
          C         CALL DPR(IPTR,ADRX,H0,NWDS)
          C1391     SVAL(NOD)=H0(ISAVE(1,NOD))
          C         NOD=NOD+1
          C         IF(NOD.GT.NOLRG) GO TO 1390
          C         IF(ISAVE(2,NOD).EQ.ILINX) GO TO 1391
          C1392     CONTINUE
          C1390     CALL DPW(ILIST,SAVADR,SVAL,256.)
          C1389     CONTINUE
0513                SAVADR=SAVADR+2.
0514                UNSADR=UNSADR+4.
0515                CALL DPW(ISTATS,STATS+ISTSOF,ITPNM,256.)
          C UPDATE THE OLD PHIS
0516                DO 595 ILNN=1,NLINES
0517                BLK=(ILNN-1)*12.+1.
0518                BLK1=BLK+9.
```

```
0519            CALL DPR(IPTR,BLK,H0,NWDS)
0520            CALL DPW(IPTR,BLK1,H0,NWDS)
0521    595     CONTINUE
        C       IF(AUTO.EQ.1) GO TO 60
        C MANUAL INTERACTION
0522            ISWV=2
0523            CALL SSWTCH(2,ISWV)
0524            IF(ISWV.NE.1) GO TO 6001
        C CHANGE VALUES
0525    87803   WRITE(6,6006)OMEGA,CONV
0526    6006    FORMAT(' CHANGE OMEGA AND CONVERGENCE'/
                1' OMEGA =',F12.2,' CONVERGENCE =',F10.5/
                1' CONTINUE (0)  NEW VALUES (1)')
0527            READ(6,7007,END=87803,ERR=87803) ICHG
0528            IF(ICHG.EQ.0) GO TO 6001
0529            WRITE(6,7008)
0530            READ(6,7009,END=87803,ERR=87803) OMEGA,CONV
0531            WRITE(6,7009) OMEGA,CONV
        C60     CONTINUE
        C STORE TIME PLANE ACCORDING TO NTPLN IN THE ANSWER FILE
0532    6001    CONTINUE
        C THIS PROCEDURE READS THE VTTFIELDS FILE AND SAVES TO THE RESULT FILE
        C IN COMPRESSED FASHION ALLOWING A 1 LINE OVERLAP PER PAGE
        C OFFSET IS IRSLTO+3.
        C NUMSAV IS THE TIME STEP NUMBER
0533            DO 27133 IPG=1,NPGS
0534            ISTLN=(IPG-1)*NLIN+1
0535            ISTPLN=ISTLN-1+NLIN
0536            IF(ISTPLN.GT.NLINES) ISTPLN=NLINES
0537            IF(IPG.NE.NPGS) ISTPLN=ISTPLN+1
0538            WDS=NCOLS*2.
0539            WRDS=WDS
0540            DO 27132 IL=ISTLN,ISTPLN
0541            ADR=(IL-1)*12.+1.
0542            IPOS=(IL-ISTLN)*NCOLS+1
0543            CALL DPR(IPTR,ADR,FA(IPOS),WRDS)
0544    27132   CONTINUE
0545            BUFWDS=(ISTPLN-ISTLN+1)*NCOLS*2.
0546            ADR=IRSLTO+3.+ITOTBK*NUMSAV+(IPG-1)*IBLKPP
0547            CALL DPW(IANS,ADR,FA,BUFWDS)
0548    27133   CONTINUE
        C
        C
        C       DO 62 IP=1,NLINES
        C       ADR=(IP-1)*12.+1.
        C       ADR1=NUMSAV*200.+IP-1.+3.
        C       CALL DPR(IPTR,ADR,H0,NWDS)
        C       CALL DPW(IANS,ADR1+IRSLTO,H0,NWDS)
        C62     CONTINUE
0549            IF(NSTEP.NE.NSTEPS(NTPLN)) GO TO 6611
0550            NTPLSV=NTPLSV+1
0551            RTIMES(1,NTPLSV)=TIMVAL(1,NTPLN)
0552            RTIMES(2,NTPLSV)=TIMVAL(2,NTPLN)*100+TIMVAL(3,NTPLN)
0553            RTIMES(3,NTPLSV)=TIMVAL(4,NTPLN)*100+TIMVAL(5,NTPLN)
0554            ISAVNO(NTPLSV)=NUMSAV
0555            PRIN(NTPLSV)=RMININ
```

```
FORTRAN IV-PLUS V02-51D          16:16:42     30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0556     6611     CONTINUE
0557              NUMSAV=NUMSAV+1
0558              CALL DPW(IANS,0.+IRSLTO,NUMSAV,602.)
0559     60       CONTINUE
0560              NUMSVR=NUMSAV
0561              TIMEE=TIME-STIME
0562              STATS0=STATS
0563              CALL DPW(IPTR,0.,RUNK,100.)
0564              CALL DPW(IRSPTR,0.+IRSTOF,LTMPLN,60.)
0565              ISWV=2
0566              CALL SSWTCH(2,ISWV)
0567              IF(ISWV.EQ.1) GO TO 7005
0568              IF(AUTO.EQ.1) GO TO 70
         C DO MANUAL INTERACTION BETWEEN TIME PLANES
0569     7005     WRITE(6,7004) (TITLE(KKK),KKK=1,71)
0570     7004     FORMAT(1X,71A1)
0571              WRITE(6,7006) RMININ,OMEGA,CONV
0572     7006     FORMAT(' TIME PLANE SAVED FOR',F12.2 ,' MINUTES INTO'
                  1,' THE SOLUTION'/' OMEGA=',F12.3,' CONVERGENCE =',
                  1F10.5/' CONTINUE (0)  NEW VALUES (1)')
0573              READ(6,7007,END=7005,ERR=7005) ICHG
0574     7007     FORMAT(I5)
0575              IF(ICHG.EQ.0) GO TO 70
0576              WRITE(6,7008)
0577     7008     FORMAT(' INPUT OMEGA AND CONVERGENCE')
0578              READ(6,7009,END=7005,ERR=7005) OMEGA,CONV
0579     7009     FORMAT(2F10.3)
0580              WRITE(6,7009) OMEGA,CONV
0581     70       CONTINUE
0582              STOP
0583              END
```

```
FORTRAN IV-PLUS V02-51D          16:22:07     30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0001             SUBROUTINE SHIFT(FA,IWIDE)
0002             DIMENSION FA(4608)
0003             DO 1 J=0,1408,128
0004             DO 1 I=1,IWIDE
0005             FA(I+3072+J)=FA(I+1536+J)
0006             FA(I+1536+J)=FA(I+J)
0007             CONTINUE
0008             RETURN
0009             END
```

```
FORTRAN IV-PLUS V02-510          16:22:12     30-MAY-79
VTT.FTN          /TR18LOCKS/WR

0001              SUBROUTINE LOCAL (N,IGHJ,IGHK,IA,IC)
0002              DOUBLE PRECISION ATIME,QTIM,QTIMP
0003              REAL K
0004              COMMON  H(7),HP(7),T(5),TP(5),K(5),BOT(5),TOP(5),S,C6,C7,CO,
                  1         A0(4),Q,QP,QTIM,QTIMP,ATIME,DTIME,DLX2,T0(4),OCEAN
                  1,STOFCT,POTDIF
0005              COMMON /TRIDG/ A(128),B(128),C(128),P(128),Z(128)
        C CALCULATE THE T'S
0006              SMUL=1.0
0007              DO  20 I=1,5
0008              BB=BOT(I)
0009              IF(K(I).GE.0.) GO TO 15
0010              IF(BB.EQ.0.) BB=1.0
0011              T(I)=ABS(K(I))*BB
0012              TP(I)=T(I)
0013              GO TO 20
0014      15      THK=H(I)-BB
0015              IF(H(I).GT.TOP(I)) THK=TOP(I)-BB
0016              IF(H(I).LT.TOP(I)) SMUL=STOFCT
0017              T(I)= K(I)*THK
0018              THKP=HP(I)-BB
0019              IF(HP(I).GT.TOP(I)) THKP=TOP(I)-BB
0020              TP(I)=K(I)*THKP
0021      20      CONTINUE
0022              TSUM=0.
        C CALCULATE THE T0I'S
0023              DO 30 I=1,4
0024              T0(I)=A0(I)*.25*(T(1)+TP(1)+T(I+1)+TP(I+1))
0025              TSUM=T0(I)+TSUM
0026      30      CONTINUE
0027              C(N)=-T0(1)
0028              A(N)=-T0(3)
0029              P(N)=-TSUM*H(1)+T0(1)*H(2)+T0(2)*(H(3)+HP(3))+T0(3)*H(4)
                  1+T0(4)*(H(5)+HP(5))
0030              PTERM=2.*DLX2*S/DTIME*SMUL
0031              FRAC=(ATIME-QTIM)/(QTIMP-QTIM)
0032              QTERM=2.*FRAC*(QP-Q)+2.*Q
        C CALCULATE COEFFICENTS
0033              COEFSM=0.
0034              COEFPR=0.
0035              COEFSM=C6+C7+CO
0036              COEFPR=C6*(H(6)+HP(6))+C7*(H(7)+HP(7))+CO*2.*OCEAN+C6*POTDIF
0037              P(N)=P(N)+COEFPR+QTERM-COEFSM*H(1)+PTERM*H(1)
0038              B(N)=PTERM+COEFSM+TSUM
0039              IF(IA.NE.2) GO TO 13
0040              P(N)=P(N)-A(N)*HP(4)
0041              A(N)=0.
0042      13      CONTINUE
0043              IF(IC.NE.2) GO TO 14
0044              P(N)=P(N)-C(N)*HP(2)
0045              C(N)=0
0046      14      CONTINUE
        D         IF(IGHJ.NE.IGHK) GO TO 3007
        D         WRITE(5,3005) N,TP,T0,TSUM,A(N),B(N),C(N),P(N),COEFSM,
        D         1COEFPR,PTERM,QTERM
```

```
FORTRAN IV-PLUS V02-51D          16:22:12     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

        D3005    FORMAT(' LOCAL SUBROUTINE N=',I5/2(5E20.8/),5E20.8/
        D        14E20.8)
        D        F=C(N)*H(2)
        D        F1=B(N)*H(1)
        D        F2=A(N)*H(4)
        D        FS=F+F1+F2
        D        DF=FS-P(N)
        D        WRITE(5,3006) F,F1,F2,FS,P(N),DF
        D3006    FORMAT(3E20.8)
        D        CLOSE(UNIT=5)
        D3007    CONTINUE
0047             RETURN
0048             END
```

```
FORTRAN IV-PLUS V02-51D          16:22:31     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

0001             SUBROUTINE TIMHRS(INPTIM,WDAYS,FRDAYS)
0002             INTEGER YEAR,DAY,HOUR
0003             DIMENSION INPTIM(5)
0004             YEAR=INPTIM(1)-1900
0005             MONTH=INPTIM(2)
0006             DAY=INPTIM(3)
0007             HOUR=INPTIM(4)
0008             MIN=INPTIM(5)
0009             CALL HAPD(DAY,MONTH,YEAR,ITIM)
0010             WDAYS=ITIM
0011             RMIN=MIN
0012             RHOUR=HOUR
0013             FRDAYS=(RHOUR+(RMIN/60.))/24.
0014             RETURN
0015             END
```

```
FORTRAN IV-PLUS V02-51D          16:22:35     30-MAY-79
VTT.FTN          /TR:BLOCK3/WR

0001             SUBROUTINE OMYHM(TIME,FRTIME,INPTIM)
0002             DIMENSION INPTIM(5)
0003             NHD=TIME
0004             CALL DATE(NHD,INPTIM(3),INPTIM(2),INPTIM(1))
0005             INPTIM(1)=INPTIM(1)+1900
0006             HRMIN=(FRTIME-NHD)*24.
0007             INPTIM(4)=HRMIN
0008             AMIN=(HRMIN-INPTIM(4))*60.+.1
0009             INPTIM(5)=AMIN
0010             RETURN
0011             END
```

```
FORTRAN IV-PLUS V02-510          16:22:39     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

0001              SUBROUTINE DATE (NHD,L,M,N)
        C
        C ROUTINE CONVERTS HAPO DATE INTO CALENDAR DATE
        C L - DAY
        C M - MONTH
        C N - YEAR
        C IF L,M,N =-1,  ERROR IN INPUT
        C
0002              DIMENSION MTABLE(12)
        C
0003              DATA MTABLE/0, 31, 59, 90, 120, 151, 181, 212, 243, 273,
                  1304, 334/
0004              IF (NHD)90,90,10
0005     10       CONTINUE
0006              IBASE=NHD+268
0007              JY=FLOAT(IBASE)/365.25
0008              JL=MOD(JY,4)
0009              JD=JY/4
0010              KT=IBASE-JD
0011              IF(KT-(JY*365))105,105,15
0012     15       CONTINUE
0013              KD=MOD(KT,365)
0014              IF(KD)110,110,20
0015     20       CONTINUE
0016              IF(JL)30,30,60
0017     30       CONTINUE
0018              IF(KD-59)50,40,60
0019     40       CONTINUE
0020              L=29
0021              GO TO 100
0022     50       CONTINUE
0023              KD=KD+1
0024     60       CONTINUE
0025              DO 70 I=1,12
0026              II=I
0027              IF(KD-MTABLE(I))80,80,70
0028     70       CONTINUE
0029              L=KD-334
0030              M=12
0031              N=44+JY
0032              GO TO 120
0033     80       CONTINUE
0034              M=II-1
0035              L=KD-MTABLE(M)
0036              N=44+JY
0037              GO TO 120
0038     90       CONTINUE
0039              L=-1
0040              M=-1
0041              N=-1
0042              GO TO 120
0043     100      CONTINUE
0044              M=2
0045              N=44+JY
0046              GO TO 120
```

```
FORTRAN IV-PLUS V02-51D          16:22:39     30-MAY-79
VTT.FTN          /TR:8LOCKS/WR

0047     105     CONTINUE
0048             L=1
0049             M=1
0050             N=44+JY
0051             GO TO 120
0052     110     CONTINUE
0053             L=31
0054             M=12
0055             N=44+JY
0056     120     CONTINUE
0057             RETURN
0058             END
```

```
FORTRAN IV-PLUS V02-51D          16:22:49     30-MAY-79
VTT.FTN          /TR:BLOCKS/WR

      C HAPO LIST FORMAT(DAY,MONTH,YEAR,HAPO DATE)*NHD=-1 IF ERROR INPUT
      C
0001            SUBROUTINE HAPO(L,M,N,NHD)
0002            DIMENSION MTABLE(12)
      C
0003            DATA MTABLE/0,31,59,90,120,151,181,212,
                1243,273,304,334/
0004            IBASE=268
0005            K=MTABLE(M)
0006            KY=N-44
0007            IF(KY) 10,20,30
0008     10     NHD=-1
0009            GO TO 60
0010     20     NHD=L+K-IBASE
0011            IF(NHD) 10,10,60
0012     30     J=KY/4
0013            LL=MOD(KY,4)
0014            IF(LL) 45,45,50
0015     40     NHD=L+K-IBASE+365*KY+J-1
0016            GO TO 60
0017     45     IF(M-3) 40,50,50
0018     50     NHD=L+K-IBASE+365*KY+J
0019     60     RETURN
0020            END
```

```
                                        .TITLE  TRIDIAGONAL PROGRAM
000000                                  .PSECT  TRIDG,RW,D,OVR,GBL
        000400                  SZ=128.*2.
000000                          A:      .BLKW   SZ
001000                          B:      .BLKW   SZ
002000                          C:      .BLKW   SZ
003000                          P:      .BLKW   SZ
004000                          Z:      .BLKW   SZ
        000000                  AA=0
        001000                  BA=SZ*2.
        002000                  CA=SZ*4.
        003000                  PA=SZ*6.
        004000                  ZA=SZ*8.
000000                                  .PSECT  TRIDAG,RW,I,CON,LCL
                                        .GLOBL  TRIDIG
000000                          W:      .BLKW   SZ
001000                          G:      .BLKW   SZ
        000000                  WA=0
        001000                  GA=SZ*2.
002000  000000                  IS0:    .WORD   0
002002  000000                  IS1:    .WORD   0
        000000                          F0=R0
        000001                          F1=R1
        000002                          F2=R2
        000003                          F3=R3
        000004                          F4=R4
        000005                          F5=R5
                                ;
                                ;
002004                          TRIDIG:
```

```
002004  017567  000002  177766          MOV     #2(R5),ISO      ; SAVE ISO
002012  017567  000004  177762          MOV     #4(R5),IS1      ; SAVE IS1
002020  016701  177754                  MOV     ISO,R1
002024  162701  000001                  SUB     #1,R1
002030  006301                          ASL     R1
002032  006301                          ASL     R1
002034  062701  000000'                 ADD     #A,R1   ; R1 CONTAINS THE ADDRESS OF A(ISO)
002040  016702  177734                  MOV     ISO,R2
002044  016703  177732                  MOV     IS1,R3
002050  160203                          SUB     R2,R3
002052  010304                          MOV     R3,R4   ; NOW R3 AND R4 CONTAIN IS1-ISO
002054  162702  000001                  SUB     #1,R2
002060  006302                          ASL     R2
002062  006302                          ASL     R2
002064  062702  000000'                 ADD     #W,R2   ; R2 CONTAINS THE ADDRESS OF W(ISO)
002070  172461  003000                  LDF     PA(R1),F0       ; P(ISO) TO F0
002074  172561  001000                  LDF     BA(R1),F1       ; B(ISO) TO F1
002100  174001                          DIVF    F1,F0   ; F0 = P(ISO)/B(ISO)
002102  174062  001000                  STF     F0,GA(R2)       ; G(ISO) = F0
002106  026767  177670  177664          CMP     IS1,ISO
002114  003003                          BGT     1$
002116  174061  004000                  STF     F0,ZA(R1)       ; IS1 .LE. ISO SO SET Z(ISO) = G(ISO)
002122  000207                          RTS     PC      ; AND RETURN
002124  172461  002000          1$:     LDF     CA(R1),F0       ; IS1 .GT. ISO SO CALCULATE W(ISO)
002130  174001                          DIVF    F1,F0   ; C(ISO)/B(ISO)
002132  174012                          STF     F0,(R2) ; SAVE W(ISO) = C(ISO)/B(ISO)
002134  062701  000004          LOOP1:  ADD     #4,R1
```

```
002140  062702  000004                  ADD     #4,R2
002144  172411                          LDF     (R1),F0 ; GET A(I)
002146  174002                          STF     F0,F2   ; SAVE FOR LATER
002150  171062  177774                  MULF    WA-4(R2),F0
002154  170700                          NEGF    F0
002156  172061  001000                  ADDF    BA(R1),F0       ; F0=QUE
002162  172561  002000                  LDF     CA(R1),F1
002166  174500                          DIVF    F0,F1
002170  174112                          STF     F1,(R2) ; W(I)=C(I)/QUE
002172  171262  000774                  MULF    GA-4(R2),F2
002176  170702                          NEGF    F2
002200  172261  003000                  ADDF    PA(R1),F2
002204  174600                          DIVF    F0,F2
002206  174262  001000                  STF     F2,GA(R2)       ; SAVE G(I)
002212  077330                          SOB     R3,LOOP1
002214  174261  004000                  STF     F2,ZA(R1)       ; SAVE Z(I)
002220  062701  177774          LOOP22: ADD     #-4,R1
002224  062702  177774                  ADD     #-4,R2
                                ;
                                ;
002230  172461  004004          LOOP2:  LDF     ZA+4(R1),F0
002234  171012                          MULF    (R2),F0
002236  170700                          NEGF    F0
002240  172062  001000                  ADDF    GA(R2),F0
002244  174061  004000                  STF     F0,ZA(R1)
002250  077415                          SOB     R4,LOOP22
002252  000207                          RTS     PC
        000001                          .END
```

```
TRIDIAGONAL PROGRAM     MACRO D1013  30-MAY-79 16:23
SYMBOL TABLE

A        000000R     002 F0     =%000000          G       001000R     003 LOOP2   002230R     003 TRIDIG  002004RG    003
AA     = 000000          F1     =%000001          GA    = 001000          LOOP22  002220R     003 W       000000R     003
B        001000R     002 F2     =%000002          IS0     002000R     003 P       003000R     002 WA    = 000000
BA     = 001000          F3     =%000003          IS1     002002R     003 PA    = 003000          Z       004000R     002
C        002000R     002 F4     =%000004          LOOP1   002134R     003 SZ    = 000400          ZA    = 004000
CA     = 002000          F5     =%000005

. ABS.   000000      000
         030000      001
TRIDG    005000      002
TRIDAG   002254      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  195 WORDS  ( 1 PAGES)
DYNAMIC MEMORY:  11935 WORDS  ( 45 PAGES)
ELAPSED TIME:  00:00:05
.LP:=DB0:TRIDIG
```

THIS PAGE INTENTIONALLY
LEFT BLANK

VTTSSX

THIS PAGE INTENTIONALLY
LEFT BLANK

```
A:      .FCTR   VTTSSX
B:      .FCTR   [1,1]FLQLIB,DLB/LB
C:      .FCTR   READX
D:      .FCTR   VTTGNS
E:      .FCTR   SYMFAC
F:      .FCTR   PRNTR
G:      .FCTR   RTHFAC
        .ROOT   A-B-*(C,D,E-G,F)
        .END
```

```
VTTSSX,LP/SH=VTTSSX,ODL/MP
ASG=TI:6,LP:5,DP2:2
/
```

```
FORTRAN IV-PLUS V02-510          16:23:36     30-MAY-79
VTTSSX.FTN        /TR:BLOCKS/WR

        C         NAME:   VTTSSZ
        C         PROGRAM: THIS PROGRAM SOLVES THE SYMMETRIC VERSION OF THE
        C                  STEADY STATE VTT MULTI AQUIFER EQUATION
        C
0001              BYTE STAT
0002              COMMON /BKA/ NE,NEF,M,N,NBF
0003              COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0004              WRITE(6,*) 'CHOLESKY VERSION OF S. S. VTT LINEAR CONVERGENCE'
0005              WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DBO:'
0006              READ(6,*) IUN
0007              IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0008              WRITE(6,106)
0009              READ(6,101) IFLD
0010              IF(IFLD.EQ.0) CALL OPFILE(2,'VTTFIELDS',2401.,NEF)
0011              IF(IFLD.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,NEF)
0012              CALL OPFILE(2,'VTTMATRIX',5000.,NE)
0013              NBF =140
0014              CALL OPR(NEF,0.,NN,8.,STAT)
0015              M =MM
0016              N =NN
0017              WRITE(6,104)
0018              READ(6,101) IG
0019              IF(IG .EQ. 0) GO TO 443
0020              WRITE(6,100)
0021              READ(6,101) M,N,ICOL,LINES,NIT,ITCT
0022      443     CALL PRNTR(0,1,0)
0023      444     WRITE(6,*) 'ENTER THE NUMBER OF ITTERATIONS'
0024              READ(6,*,END=445) NIT
0025      20      DO 10 IT =1,NIT
0026              IF(IT.NE.1) DTOSS=SECNDS(TOSS)/60.
0027              ITOSS=IT-1
0028              IF(IT.NE.1) WRITE(6,107) DTOSS,ITOSS
0029      107     FORMAT(' TIME=',F10.2,' MIN. FOR ITERATION=',I5)
0030              WRITE(6,102) IT
0031              TOSS=SECNDS(0.)
0032              CALL VTTGNS
0033              CALL SYMFAC
0034              CALL FBSOLV
0035              CALL READX
0036              ITCT =ITCT +1
0037      10      CONTINUE
0038              DTOSS=SECNDS(TOSS)/60.
0039              WRITE(6,107) DTOSS,NIT
0040              WRITE(6,105) NIT
0041              CALL PRNTR(0,1,0)
0042              WRITE(6,*) 'ARE YOU DONE? 0=NO 1=YES'
0043              READ(6,101) IANS
0044              IF(IANS.EQ.0) GO TO 444
0045      445     STOP
0046      100     FORMAT(1X,'INPUT M,N,ICOL,LINES,NIT,ITCT:6I5')
0047      101     FORMAT(6I6)
0048      104     FORMAT(1X,' TTY HEADER INPUT ?')
0049      102     FORMAT(1X,'START OF ITERATION',I3)
0050      105     FORMAT(1X,'FINISHED :',I4,2X,'ITERATIONS')
0051      106     FORMAT(1X,'LARGE OR SMALL FIELD ? :0=LG,1=SM')
0052              END
```

```
FORTRAN IV-PLUS V02-510          16:24:00     30-MAY-79
READX1.FTN       /TR:BLOCKS/WR

        C        ***** READX1 *****
        C        DOESN'T BUMP BOTTOM
0001             SUBROUTINE READX
0002             BYTE STAT
0003             COMMON /BKA/ NE,NEF,M,N,NBF
0004             COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005             DIMENSION RBUF(64,64),BUFF(128,8),H(128),NNO(256),BTM(128)
                 1,HYDC(128)
0006             EQUIVALENCE (BUFF(1,1),H),(BUFF(1,6),NNO),(BUFF(1,2),BTM)
                 1,(BUFF(1,3),HYDC)
0007             NLN =0
0008             DO 20 ILN =1,LINES
0009             RB =(ILN-1)*12 +1
0010             CALL DPR(NEF,RB,BUFF,8.*256.)
0011             NODE =NNO(1)
0012             IF(NODE .GT. 62) NODE =62
0013             IF(NODE .EQ. 0) GO TO 20
0014             RBF =2.*(NLN/2)
0015             IODD =MOD(NLN,2)
0016             CALL DPR(NE,RBF/2.,RBUF,(NODE+IODD)*128.)
0017             DO 5 IT =1,NODE
0018             INDX =NNO(IT+1)
0019             H(INDX) =RBUF(64,IT+IODD)
0020     5       CONTINUE
0021             IND =NNO(2)
0022             IF(HYDC(IND) .LT. 0.) GO TO 10
0023             DO 10 I =1,NODE
0024             IP =NNO(I+1)
0025             H0 =H(IP)
0026             BT0 =BTM(IP)
0027             IF(H0 .GT. BT0+.001) GO TO 10
0028             H(IP)=BT0+.001
0029             WRITE(5,100) ILN,IP,H0,BT0,H(IP)
0030     10      CONTINUE
0031             CALL DPW(NEF,RB,BUFF,2048.)
0032             NLN =NLN +NODE
0033     20      CONTINUE
0034             CLOSE(UNIT=5)
0035             RETURN
0036     100     FORMAT(1X,'POTENTIAL PROB AT LINE =',I4,'  COLL=',I4,5X
                 1,'PHE =',F10.2,5X,' BOTTOM =',F10.2,5X,'NEW POTENTIAL =',F10.2)
0037             END
```

```
FORTRAN IV-PLUS V02-510          16:24:25     30-MAY-79
VTTGNS.FTN        /TR:8LOCKS/WR

        C         ***** VTTGNS *****
        C         PROGRAM: SUBROUTINE FOR GENERATING SYMMETRIC 3 AQUIFER
        C                  CONFINED - UNCONFINED MATRIX
        C
0001              SUBROUTINE VTTGNS
0002              BYTE STAT
0003              COMMON /BKA/ NE,NEF,M,N,NBF
0004              COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005              DIMENSION TRBUF(64,80),ACOFFR(4,27)
0006              REAL H2(128),H0(128),H4(128),BTM2(128),BTM0(128),BTM4(128)
                  1,K2(128),K0(128),K4(128),Q0(128),TRN0(128),KEF
                  2,TOP2(128),TOP0(128),TOP4(128)
0007              INTEGER LNBF(256,24),NTP2(256),NTP0(256),NTP4(256),NNO2(256)
                  1,NNO0(256),NNO4(256),INDX3(4,3),IHOD(12)
0008              EQUIVALENCE (LNBF(1,1),H2),(LNBF(1,9),H0),(LNBF(1,17),H4)
                  1,(LNBF(1,2),BTM2),(LNBF(1,10),BTM0),(LNBF(1,18),BTM4)
                  2,(LNBF(1,3),K2),(LNBF(1,11),K0),(LNBF(1,19),K4)
                  3,(LNBF(1,12),Q0),(LNBF(1,5),NTP2),(LNBF(1,13),NTP0)
                  4,(LNBF(1,21),NTP4),(LNBF(1,6),NNO2),(LNBF(1,14),NNO0)
                  5,(LNBF(1,22),NNO4),(LNBF(1,16),TRN0),(IHOD(11),NAQF)
                  6,(LNBF(1,7),TOP2),(LNBF(1,15),TOP0),(LNBF(1,23),TOP4)
0009              DIMENSION H(7),BTM(5),K(5),NTP(6),NNO(7),CN(7),TN(5),AA(4)
0010              DIMENSION TAV(4),TOP(5)
0011              REAL K
0012              DATA ACOFFR/1.,1.,1.,1. ,0.,0.,0.,0. ,0.,0.,0.,0.
                  1, .5,1.,.5,0. ,0.,1.,1.,0. ,0.,.5,1.,.5
                  2 ,0.,0.,1.,1., .5,0.,.5,1. ,1.,0.,0.,1.
                  3 ,1.,.5,0.,.5 ,1.,1.,0.,0. ,0.,1.,.5,0.
                  5 ,0.,.5,1.,0. ,0.,0.,1.,.5 ,0.,0.,.5,1.
                  6 ,.5,0.,0.,1., 1.,0.,0.,.5 ,1.,.5,0.,0.
                  7 ,.5,1.,0.,0. ,1.,0.,.5,1. ,1.,.5,0.,1.
                  8 ,1.,1.,0.,.5 ,1.,1.,.5,0. ,.5,1.,1.,0.
                  9 ,0.,1.,1.,.5 ,0.,.5,1.,1., .5,0.,1.,1./
0013              DATA INDX3/1,2,0,2,  -1,1,-1,0,  -2,-1,0,-1/
        C
        C
0014              CALL DPR(NEF,0.,IHOD(1),12.)
0015              ICOL3 =ICOL/NAQF
0016              IWBF =80
0017              IWCT =0
0018              H(6) =0.
0019              H(7) =0.
0020              TR5 =0.
0021              TR6 =0.
0022              NTP(5) =1
0023              NTP(6) =1
0024              ITR =1
0025    1         DO 5 I =1,M
0026              DO 5 J =1,8
0027    5         LNBF(I,J) =0.
0028              CALL DPR(NEF,1.,H0,8.,*256.)
0029              CALL DPR(NEF,13.,H4,8.,*256.)
0030              DO 70 ILN =1,LINES
0031              NDPTS =NNO0(1)
0032              IF(NDPTS .EQ. 0) GO TO 60
0033              DO 60 INODE =1,NDPTS
```

```
0034            IP =NNO0(INODE+1)
0035            IAQF =(IP-1)/ICOL3 +1
0036            IH5 =INDX3(1,IAQF)*ICOL3 +IP
0037            IH6 =INDX3(2,IAQF)*ICOL3 +IP
0038            IT5 =INDX3(3,IAQF)*ICOL3 +IP
0039            IT6 =INDX3(4,IAQF)*ICOL3 +IP
0040            H(1) =H0(IP)
0041            BTM(1) =BTM0(IP)
0042            TOP(1) =TOP0(IP)
0043            K(1) =K0(IP)
0044            NTP(1) =NTP0(2*IP+2)
0045            NNO(1) =NTP0(2*IP-1)
0046            H(2) =H0(IP+1)
0047            BTM(2) =BTM0(IP+1)
0048            TOP(2) =TOP0(IP+1)
0049            K(2) =K0(IP+1)
0050            NTP(2) =NTP2(2*IP)
0051            NNO(2) =NTP0(2*IP+1)
0052            H(3) =H2(IP)
0053            BTM(3) =BTM2(IP)
0054            TOP(3) =TOP2(IP)
0055            K(3) =K2(IP)
0056            NNO(3) =NTP2(2*IP-1)
0057            IF(IP .EQ. 1) GO TO 11
0058            H(4) =H0(IP-1)
0059            BTM(4) =BTM0(IP-1)
0060            TOP(4) =TOP0(IP-1)
0061            K(4) =K0(IP-1)
0062            NTP(3) =NTP0(2*IP-2)
0063            NNO(4) =NTP0(2*IP-3)
0064     11     H(5) =H4(IP)
0065            BTM(5) =BTM4(IP)
0066            TOP(5) =TOP4(IP)
0067            K(5) =K4(IP)
0068            NTP(4) =NTP4(2*IP)
0069            NNO(5) =NTP4(2*IP-1)
0070            NDTYP =NTP0(2*IP)
0071            GO TO (201,202,203),NAQF
0072     203    H(7) =H0(IH6)
0073            NTP(6) =NTP0(2*IH6)
0074            NNO(7) =NTP0(2*IH6-1)
0075            TR6 =TRN0(IT6)
0076     202    H(6) =H0(IH5)
0077            NTP(5) =NTP0(2*IH5)
0078            NNO(6) =NTP0(2*IH5-1)
0079            TR5 =TRN0(IT5)
0080     201    IF(NDTYP .GT. 2) NDTYP =NDTYP-5
0081            DO 10 I =1,4
0082     10     AA(I) =ACOFFR(I,NDTYP+1)
         C
         C                        COMPUTE COFF.
         C
0083            DO 15 I =1,5
0084            BB =BTM(I)
0085            IF(K(I) .GE. 0) GO TO 12
0086            IF(BB .EQ. 0.) BB =1.
```

```
0087            TN(I) =ABS(K(I))*BB
0088            GO TO 15
0089     12     THICK =H(I) -BB
0090            IF(H(I) .GT. TOP(I)) THICK =TOP(I) -BB
0091            TN(I) =K(I)*THICK
0092     15     CONTINUE
0093            DO 20 I =1,4
0094     20     TAV(I) =(TN(I) +TN(I+1))/2.
0095            DO 25 I =1,4
0096            CN(I+1) =AA(I)*TAV(I)
0097            IF(NTP(I) .EQ. 2) CN(I+1) =0.
0098     25     CONTINUE
0099            IF(NTP(5) .EQ. 1) TR5 =0.
0100            IF(NTP(6) .EQ. 1) TR6 =0.
0101            CN(6) =TR5
0102            CN(7) =TR6
0103            IF(NTP(5) .EQ. 2) CN(6) =0.
0104            IF(NTP(6) .EQ. 2) CN(7) =0.
0105            SM3 =0.
0106            DO 30 I =1,4
0107            SM3 =SM3 +AA(I)*TAV(I)
0108     30     CONTINUE
0109            CN(1) =-SM3 -TR5 -TR6
0110            F0 =0.
0111            DO 35 I =1,4
0112            F0 =F0 +AA(I)*TAV(I)*(H(I+1) -H(1))
0113     35     CONTINUE
0114            F0 =F0 + Q0(IP) +TR5*(H(6) -H(1)) +TR6*(H(7) -H(1))
0115            BTRM =0.
0116            DO 40 I =1,4+NAQF
0117     40     BTRM =BTRM +CN(I)*H(I)
0118            BTRM =BTRM -F0
0119     41     CONTINUE
         C
         C                          STORE COFF IN MARTIX
         C
0120            DO 45 I =1,M+1
0121     45     TRBUF(I,ITR) =0.
0122            TRBUF(64,ITR) =0.
0123            DO 50 I =1,4+NAQF
0124            IF(CN(I) .EQ. 0) GO TO 50
0125            INDX =1 +NNO(I) -NNO(1)
0126            IF(INDX .GT. M+1) GO TO 50
0127            IF(INDX .LT. 1) GO TO 50
0128            TRBUF(INDX,ITR) = -CN(I)
0129     50     CONTINUE
0130            TRBUF(64,ITR) = -BTRM
0131            IF(NNO(1) .EQ. N) GO TO 55
0132            IF(ITR .EQ. IWBF) GO TO 55
0133            ITR =ITR +1
0134            GO TO 60
0135     55     WRLC =IWCT/2
0136            CALL DPW(NE,WRLC,TRBUF,ITR*128.)
0137            ITR =1
0138            IWCT =IWCT +IWBF
0139     60     CONTINUE
```

```
FORTRAN IV-PLUS V02-S1D          16:24:25     30-MAY-79
VTTGNS.FTN       /TR:8LOCKS/WR

        C
        C                        SHIFT AND READ
        C
0140            DO 65 I =1,2*ICOL
0141            DO 65 J =1,16
0142            LNBF(I,J) =LNBF(I,J+8)
0143    65      CONTINUE
0144            RRD =(ILN +1)*12 +1
0145            IF(ILN +2 .GT. LINES) GO TO 70
0146            CALL DPR(NEF,RRD,H4,8,*256,)
0147    70      CONTINUE
0148            RETURN
0149            END
```

```
NE
FORTRAN IV-PLUS V02-51D          16:25:34     30-MAY-79
SYMFAC.FTN       /TR:BLOCKS/WR

        C        ***** SYMFAC *****
0001             SUBROUTINE SYMFAC
0002             COMMON /BKA/ NE,NEF,M,NTOT,NBF
0003             COMMON /BUFF/ A(64,140)
0004             CALL DPR(NE,0.,A,NBF*128.)
0005             CALL SEGFAC(1,1,NBF)
0006             NSOLV =2*((NBF -M)/2)
0007             KSEG =0
0008     1       KSEG =KSEG +1
0009             RLW =(KSEG-1)*NSOLV
0010             NSTRT =KSEG*NSOLV +1
0011             CALL DPW(NE,RLW/2.,A(1,1),NSOLV*128.)
0012             CALL SHFTUP(NSOLV)
0013             RLR =RLW +NBF
0014             NLST =RLR +NSOLV
0015             IRD =NBF -NSOLV +1
0016             IF(NLST .GE. NTOT) GO TO 5
0017             CALL DPR(NE,RLR/2.,A(1,IRD),NSOLV*128.)
0018             CALL SEGFAC(NSTRT,IRD,NSOLV)
0019             GO TO 1
0020     5       NOVER =NLST -NTOT
0021             NRED =NSOLV -NOVER
0022             WBK =NSTRT -1
0023             NSOLV =NBF -NOVER
0024             CALL DPR(NE,RLR/2.,A(1,IRD),NRED*128.)
0025             CALL SEGFAC(NSTRT,IRD,NSOLV)
0026             CALL DPW(NE,WBK/2.,A,(NBF-NOVER)*128.)
0027             RETURN
0028             END
```

```
FORTRAN IV-PLUS V02-51D          16:26:06     30-MAY-79
SYMFAC.FTN       /TR:BLOCKS/WR

0001             SUBROUTINE FBSOLV
0002             COMMON /BKA/ NE,NEF,M,NTOT,NBF
0003             COMMON /BUFF/ A(64,140)
0004             CALL DPR(NE,0.,A,NBF*128.)
0005             NSOLV =2*((NBF -M)/2)
        C
        C                FORWARD SUBSTITUTION
        C
0006             KSEG =0
0007     1       KSEG =KSEG +1
0008             RLW =(KSEG-1)*NSOLV
0009             NSTRT =RLW +1
0010             CALL SEGFWD(NSTRT,NSOLV)
0011             CALL DPW(NE,RLW/2.,A,NSOLV*128.)
0012             CALL SHFTUP(NSOLV)
0013             RLR =RLW +NBF
0014             NLST =RLR +NSOLV
0015             IRD =NBF -NSOLV +1
0016             IF(NLST .GE. NTOT) GO TO 5
0017             CALL DPR(NE,RLR/2.,A(1,IRD),NSOLV*128.)
0018             GO TO 1
0019     5       NOVER =NLST -NTOT
0020             NRED =NSOLV -NOVER
0021             NSTRT =RLW +NSOLV +1
0022             WBK =NSTRT -1
0023             NSOLV =NBF -NOVER
0024             CALL DPR(NE,RLR/2.,A(1,IRD),NRED*128.)
0025             CALL SEGFWD(NSTRT,NSOLV)
0026             CALL DPW(NE,WBK/2.,A,NSOLV*128.)
        C
        C                BACKWARD SUBSTITUTION
        C
0027             KSEG =0
0028             NSOLV =2*((NBF -M)/2)
0029             RL =2*((NTOT +1)/2) -NBF
0030             CALL DPR(NE,RL/2.,A,NBF*128.)
0031    10       KSEG =KSEG +1
0032             NOD =0
0033             IF(KSEG .EQ. 1) NOD =MOD(NTOT,2)
0034             RLW =2*((NTOT +1)/2) -KSEG*NSOLV
0035             NSTRT =RLW +NSOLV
0036             IRD =NBF -NSOLV +1
0037             CALL SEGBWD(NSTRT-NOD,NBF-NOD,NSOLV-NOD)
0038             CALL DPW(NE,RLW/2.,A(1,IRD),NSOLV*128.)
0039             CALL SFTDWN(NSOLV)
0040             RLR =RLW -NBF
0041             IF(RLR .LE. 0) GO TO 15
0042             CALL DPR(NE,RLR/2.,A,NSOLV*128.)
0043             GO TO 10
0044    15       NSOLV =NBF +RLR
0045             NSTRT =NBF
0046             RED =2*((NBF -M)/2) +RLR
0047             JA =1 -RLR
0048             CALL DPR(NE,0.,A(1,JA),RED*128.)
0049             CALL SEGBWD(NSTRT,NBF,NSOLV)
0050             CALL DPW(NE,0.,A(1,JA),NSOLV*128.)
0051             RETURN
0052             END
```

```
FORTRAN IV-PLUS V02-510          16:26:33     30-MAY-79
PRNTR.FTN        /TR:8LOCKS/WR

        C         ****** SUBROUTINE PRNTR ******
0001              SUBROUTINE PRNTR(LWRAQ,IMETH,IOLD)
0002              BYTE STAT,FLG(2)
0003              COMMON /BKA/ NE,NEF,M,N,NBF
0004              COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005              DIMENSION BUFF(128),ACOFF(4,27),ACOFFR(4,27)
0006              REAL H2(128),H0(128),H4(128),BTM2(128),BTM0(128),BTM4(128)
                  1,K2(128),K0(128),K4(128),Q0(128),TRN0(128),TOP2(128),BUFB(64,2)
                  2,TOP0(128),TOP4(128),OCN0(128),LEAK,LEAKK
0007              INTEGER LNBF(256,27),NTP2(256),NTP0(256),NTP4(256),NNO2(256)
                  1,NNO0(256),NNO4(256),IHDR(90),IPP(128),INDX3(4,3)
0008              EQUIVALENCE (LNBF(1,1),H2),(LNBF(1,2),BTM2),(LNBF(1,3),K2)
                  1,(LNBF(1,5),NTP2),(LNBF(1,7),TOP2),(LNBF(1,10),H0)
                  2,(LNBF(1,11),BTM0),(LNBF(1,12),K0),(LNBF(1,13),Q0)
                  3,(LNBF(1,14),NTP0),(LNBF(1,15),NNO0),(LNBF(1,16),TOP0)
                  4,(LNBF(1,17),TRN0),(LNBF(1,18),OCN0),(LNBF(1,19),H4)
                  5,(LNBF(1,20),BTM4),(LNBF(1,21),K4),(LNBF(1,23),NTP4)
                  6,(LNBF(1,25),TOP4),(BUFF,BUFB)
0009              EQUIVALENCE (IHDR(44),FLG),(IHDR(87),LEAK),(IHDR(89),HSEA)
                  1,(IHDR(9),BIAS),(IHDR(11),NAQF)
0010              DIMENSION H(7),BTM(5),K(5),NTP(6),TN(5),AA(4),TOP(5),NNO(7)
0011              DIMENSION TAV(6),AREA(4)
0012              REAL K,KEF
0013              DATA ACOFFR/1.,1.,1.,1. ,0.,0.,0.,0. ,0.,0.,0.,0.
                  1, .5,1.,.5,0. ,0.,1.,1.,0. ,0.,.5,1.,.5
                  2 ,0.,0.,1.,1., .5,0.,.5,1. ,1.,0.,0.,1.
                  3 ,1.,.5,0.,.5 ,1.,1.,0.,0. ,0.,1.,.5,0.
                  5 ,0.,.5,1.,0. ,0.,0.,1.,.5 ,0.,0.,.5,1.
                  6 ,.5,0.,0.,1., 1.,0.,0.,.5 ,1.,.5,0.,0.
                  7 ,.5,1.,0.,0. ,1.,0.,.5,1. ,1.,.5,0.,1.
                  8 ,1.,1.,0.,.5 ,1.,1.,.5,0. ,.5,1.,1.,0.
                  9 ,0.,1.,1.,.5 ,0.,.5,1.,1., .5,0.,1.,1./
0014              DATA INDX3/1,2,0,2,  -1,1,-1,0,  -2,-1,0,-1/
0015              DATA AREA/1.,.5,.375,.625/
        C
        C
0016    1         WRITE(6,100)
0017              READ(6,101,END=1001) IP
0018              IF(IP .EQ. 0) RETURN
0019              IF(IP.NE.0) GO TO 1000
0020    1001      STOP
0021    1000      CONTINUE
0022              DO 2 I =1,128
0023    2         IPP(I) =I
0024              CALL DPR(NEF,0.,IHDR,90.)
0025              LEAKK=LEAK
0026              IF(FLG(2) .EQ. 0) LEAK =0.
        C         DEEPLV=BIASS
0027              ICOL3 =ICOL/NAQF
0028              H(6) =0.
0029              H(7) =0.
0030              TR5 =0.
0031              TR6 =0.
0032              NTP(5) =1
0033              NTP(6) =1
0034              WRITE(5,102)
```

```
FORTRAN IV-PLUS V02-51D          16:26:33     30-MAY-79
PRNTR.FTN        /TR:BLOCKS/WR

0035             WRITE(5,103) NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0036             WRITE(5,107)
        C
        C                          PRINT MATRIX
        C
0037             WRITE(6,104)
0038             READ(6,101) IP
0039             IF(IP .EQ. 0) GO TO 20
0040     3       WRITE(6,105)
0041             READ(6,101) IST,IND
0042             IF(IST .EQ. 0) GO TO 20
0043             IW =M+2
0044             IF(IMETH.EQ.2) IW=2*M+2
0045             INPT =IW/12
0046             IF(MOD(IW,12) .NE. 0) INPT =INPT +1
0047             DO 15 ICL =1,INPT
0048             I1 =(ICL-1)*12 +1
0049             I2 =I1 +11
0050             WRITE(5,111)(IPP(J),J =I1,I2)
0051             DO 10 ILL =IST,IND
0052             IF(IMETH.EQ.2) GO TO 1113
0053             RL =2.*((ILL-1)/2)
0054             CALL DPR(NE,RL/2.,BUFB,256.)
0055             ILA =MOD(ILL-1,2) +1
0056             BUFB(M+2,ILA) =BUFB(64,ILA)
0057             WRITE(5,109)ILL,(BUFB(J,ILA),J =I1,I2)
0058             GO TO 10
0059     1113    CONTINUE
0060             RL=ILL-1
0061             CALL DPR(NE,RL,BUFF,256.,STAT)
0062             WRITE(5,109)ILL,(BUFF(J),J=I1,I2)
0063     10      CONTINUE
0064             WRITE(5,107)
0065     15      CONTINUE
0066             CLOSE(UNIT=5)
0067             GO TO 3
        C
        C                          PRINT SOLUTION OR THICKNESS
        C
0068     20      WRITE(6,108)
0069             READ(6,101) IPI
0070             IF(IPI .EQ. 0) GO TO 40
0071             IST =ICOL/12
0072             IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0073             DO 30 ICL =1,IST
0074             I1 =(ICL-1)*12 +1
0075             I2 =I1 +11
0076             IF(IPI .EQ. 1) WRITE(5,114)BIAS
0077             IF(IPI .EQ. 2) WRITE(5,115)
0078             IF(IPI .EQ. 3) WRITE(5,118)
0079             IF(IPI.EQ.4) WRITE(5,126)
0080     126     FORMAT(55X,'STRESS (CFD)'/)
0081             WRITE(5,111)(IPP(J),J =I1,I2)
0082             QIOT=0.
0083             DO 25 ILNS =LINES,1,-1
0084             RL =(ILNS-1)*12 +1
```

```
0085              CALL DPR(NEF,RL,H0,8,*256,,STAT)
0086              IF(IPI.NE.4) GO TO 21333
0087              IF(ICL.NE.1) GO TO 21333
0088              DO 21332 IU=1,ICOL
0089     21332    QTOT=QTOT+H0(IU)
0090              QGPM=QTOT/192.5
0091     21333    CONTINUE
0092              GO TO (35,21,24,24),IPI
0093     21       DO 22 IP =I1,I2
0094              THICK =H0(IP) -BTM0(IP)
0095              IF(H0(IP) .GT. TOP0(IP)) THICK =BTM0(IP) -TOP0(IP)
0096              BUFF(IP) =THICK
0097     22       CONTINUE
0098              GO TO 27
0099     35       DO 36 IP =I1,I2
0100     36       BUFF(IP) =H0(IP)
0101              GO TO 27
0102     24       DO 26 IP =I1,I2
0103              OFF =BIAS
0104              IF(NTP0(2*IP) .EQ. 1) OFF =0.
0105              IF(IP .GT. ICOL) OFF =0.
0106              BUFF(IP) =H0(IP) -OFF
0107              IF(IPI.EQ.4) BUFF(IP)=Q0(IP)
0108     26       CONTINUE
0109     27       IF(IPI.NE.4)WRITE(5,106)ILNS,(BUFF(J),J =I1,I2)
0110              IF(IPI.EQ.4)WRITE(5,166)ILNS,(BUFF(J),J=I1,I2)
0111     166      FORMAT(1X,I5,2X,12F10.1)
0112     25       CONTINUE
0113              IF(IPI.EQ.4.AND.ICL.EQ.IST) WRITE(5,*) 'TOTAL STRESS GPM=',
                  1QGPM
0114              WRITE(5,107)
0115     30       CONTINUE
0116              CLOSE(UNIT=5)
0117              GO TO 20
         C
         C                          PRINT FLOW BALANCE
         C
0118     40       WRITE(6,110)
0119              READ(6,101) IPT
0120              IF(IPT .EQ. 0) GO TO 300
0121              IST =ICOL/12
0122              IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0123              QTOTAL =0.
0124              DO 50 ICL =1,IST
0125              I1 =(ICL-1)*12 +1
0126              I2 =I1 +11
0127              IF(IPT .EQ. 1) WRITE(5,116)
0128              IF(IPT .EQ. 2) WRITE(5,119)
0129              IF(IPT .EQ. 3) WRITE(5,122)
0130              IF(IPT .EQ. 4) WRITE(5,123)
0131              IF(IPT.EQ.5) WRITE(5,125)
0132              WRITE(5,113)(IPP(J),J =I1,I2)
0133              DO 70 ILN =LINES,2,-1
0134              RR =(ILN -2)*12 +1
0135              CALL DPR(NEF,RR,H2,9,*256,)
0136              CALL DPR(NEF,RR+12,,H0,9,*256,)
```

```
0137            CALL DPR(NEF,RB+24.,H4,9,*256.)
0138            DO 44 IZ =I1,I2
0139     44     BUFF(IZ) =0.
0140            IF(NNO0(1) .EQ. 0) GO TO 60
0141            DO 60 IP =I1,I2
0142            IF(IP .GT. ICOL) GO TO 60
0143            F0 =0.
0144            NTPP =NTP0(2*IP)
0145            IF(NTPP .EQ. 1 .OR. NTPP .EQ. 2) GO TO 60
0146            H(1) =H0(IP)
0147            IAQF =(IP-1)/ICOL3 +1
0148            IH5 =INDX3(1,IAQF)*ICOL3 +IP
0149            IH6 =INDX3(2,IAQF)*ICOL3 +IP
0150            IT5 =INDX3(3,IAQF)*ICOL3 +IP
0151            IT6 =INDX3(4,IAQF)*ICOL3 +IP
0152            LEAK=0.
0153            CTRN=0.
0154            DEEP=0.
0155            IF(.NOT.(IAQF.EQ.1.AND.FLG(2).EQ.1))GO TO 1089
0156            CTRN=OCN0(IP)
        C       DEEP=TRN0(IP)-DEEPLV+H0(IH5)
0157            DEEP=TRN0(IP)+(H0(IH5)-TRN0(IH5))
0158            LEAK=LEAKK
0159     1089   CONTINUE
0160            IF(FLG(2).EQ.0) LEAK=0.
0161            IF(LWRAQ.EQ.0) LEAK=0.
0162            H(1) =H0(IP)
0163            BTM(1) =BTM0(IP)
0164            TOP(1) =TOP0(IP)
0165            K(1) =K0(IP)
0166            NTP(1) =NTP0(2*IP+2)
0167            NNO(1) =NTP0(2*IP-1)
0168            H(2) =H0(IP+1)
0169            BTM(2) =BTM0(IP+1)
0170            TOP(2) =TOP0(IP+1)
0171            K(2) =K0(IP+1)
0172            NTP(2) =NTP2(2*IP)
0173            NNO(2) =NTP0(2*IP+1)
0174            H(3) =H2(IP)
0175            BTM(3) =BTM2(IP)
0176            TOP(3) =TOP2(IP)
0177            K(3) =K2(IP)
0178            NNO(3) =NTP2(2*IP-1)
0179            IF(IP .EQ. 1) GO TO 11
0180            H(4) =H0(IP-1)
0181            BTM(4) =BTM0(IP-1)
0182            TOP(4) =TOP0(IP-1)
0183            K(4) =K0(IP-1)
0184            NTP(3) =NTP0(2*IP-2)
0185            NNO(4) =NTP0(2*IP-3)
0186     11     H(5) =H4(IP)
0187            BTM(5) =BTM4(IP)
0188            TOP(5) =TOP4(IP)
0189            K(5) =K4(IP)
0190            NTP(4) =NTP4(2*IP)
0191            NNO(5) =NTP4(2*IP-1)
```

```
0192            NDTYP =NTP0(2*IP)
0193            ARF=AREA(NDTYP/8+1)
0194            IF(NDTYP.EQ.1) ARF=0.
0195            GO TO (201,202,203),NAQF
0196    203     H(7) =H0(IH6)
0197            NTP(6) =NTP0(2*IH6)
0198            NNO(7) =NTP0(2*IH6-1)
0199            TR6 =TRN0(IT6)
0200    202     H(6) =H0(IH5)
0201            NTP(5) =NTP0(2*IH5)
0202            NNO(6) =NTP0(2*IH5-1)
0203            TR5 =TRN0(IT5)
0204    201     IF(NDTYP .GT. 2) NDTYP =NDTYP-5
0205            IF(FLG(2).EQ.0) GO TO 2015
0206            H(7)=0.
0207            NTP(5)=1
0208            NTP(6)=1
0209            NNO(7)=0
0210            NNO(6)=0
0211            TR5=0.
0212            TP6=0.
0213            H(6)=0.
0214            IF(.NOT.(IAQF.EQ.1)) GO TO 2014
0215            IF(LWRAQ.EQ.0) GO TO 2015
        C       H(6)=H0(IH5)-DEEPLV+TRN0(IP)
0216            H(6)=TRN0(IP)+(H0(IH5)-TRN0(IH5))
0217            NNO(6)=NTP0(2*IH5-1)
0218            NTP(5)=NTP0(2*IH5)
0219            ARFF=AREA(NTP(5)/8+1)
0220            IF(NTP(5).EQ.1) ARFF=0.
0221            ARF=MIN(ARFF,ARF)
0222            TR5=LEAKK*ARF
0223            GO TO 2015
0224    2014    CONTINUE
0225            IF(.NOT.(IAQF.EQ.2)) GO TO 2015
0226            IF(LWRAQ.EQ.0) GO TO 2015
        C       H(6)=H0(IH5)+DEEPLV-TRN0(IH5)
0227            H(6)=H0(IH5)+TRN0(IP)-TRN0(IH5)
0228            NNO(6)=NTP0(2*IH5-1)
0229            NTP(5)=NTP0(2*IH5)
0230            ARFF=AREA(NTP(5)/8+1)
0231            IF(NTP(5).EQ.1) ARFF=0.
0232            ARF=MIN(ARFF,ARF)
0233            TR5=LEAKK*ARF
0234    2015    CONTINUE
0235            DO 510 I =1,4
0236    510     AA(I) =ACOFFR(I,NDTYP+1)
0237            DEEPOF=0.
0238            TR8=0.
0239            IF(IOLD.NE.1) GO TO 3015
0240            POTDIF=0.
0241            CTRN=0.
0242            TR5=0.
0243            TP6=0.
0244            IF(IAQF.NE.1) GO TO 3015
0245            CTRN=UCN0(IP)
```

```
0246            IF(LWRAQ.EQ.0) GO TO 3015
0247            DEEPDF=TRN0(IP)-H(1)
0248            TR8=LEAKK*ARF
0249    3015    CONTINUE
        C
        C                          COMPUTE NET Q ,OR INT AQUIFER TRANSFER
        C
0250            DO 515 I =1,5
0251            BB =BTM(I)
0252            IF(K(I) .GE. 0) GO TO 512
0253            IF(BB .EQ. 0.) BB =1.
0254            TN(I) =ABS(K(I))*BB
0255            GO TO 515
0256    512     THICK =H(I) -BB
0257            IF(H(I) .GT. TOP(I)) THICK =TOP(I) -BB
0258            TN(I) =K(I)*THICK
0259    515     CONTINUE
0260            DO 520 I =1,4
0261    520     TAV(I) =(TN(I) +TN(I+1))/2.
0262            IF(NTP(5) .EQ. 1) TR5 =0.
0263            IF(NTP(6) .EQ. 1) TR6 =0.
0264            F0 =0.
0265            QMAG =0.
0266            DO 535 I =1,4
0267            FLUX = AA(I)*TAV(I)*(H(I+1) -H(I))
0268            F0 =F0 +FLUX
0269            QMAG =QMAG +ABS(FLUX)
0270    535     CONTINUE
0271            ADIV =0.
0272            DO 540 I =1,4
0273    540     ADIV =ADIV +AA(I)
0274            TRNF =TR5*(H(6) -H(1)) +  TR6*(H(7) -H(1))+TR8*DEEPDF
0275            OCNXF=CTRN*(HSEA-H(1))
0276            IF(ADIV .EQ. 0.) GO TO 60
0277            ABFLX =(QMAG )/ADIV
0278            IF(ABFLX .EQ. 0.) GO TO 60
0279            FLRAT =100.*ABS(F0/ABFLX)
0280            IF(IPT .EQ. 1) BUFF(IP) =-F0 -TRNF-OCNXF
0281            IF(IPT .EQ. 2) BUFF(IP) =TRNF
0282            IF(IPT .EQ. 3) BUFF(IP) =ABFLX
0283            IF(IPT .EQ. 4) BUFF(IP) =FLRAT
0284            IF(IPT.EQ.5) BUFF(IP)=OCNXF
0285            QTOTAL =QTOTAL +BUFF(IP)
0286    60      CONTINUE
0287            IF(IPT .NE. 4) WRITE(5,109) ILN,(BUFF(J),J =I1,I2)
0288            IF(IPT .EQ. 4) WRITE(5,124) ILN,(BUFF(J),J =I1,I2)
0289    70      CONTINUE
0290            WRITE(5,107)
0291    50      CONTINUE
0292            GPM =QTOTAL/192.5
0293            IF(IPT.NE.4)WRITE(5,121) GPM
0294            WRITE(5,107)
0295            CLOSE(UNIT=5)
0296            GO TO 40
        C
        C                          PRINT TRANSMISSIVITY
```

```
      C
0297  300     WRITE(6,112)
0298          READ(6,101) IPT
0299          IF(IPT .EQ. 0) RETURN
0300          CNVT =1.
0301          IF(IPT .EQ. 2) CNVT =7.481
0302          IST =ICOL/12
0303          IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0304          DO 330 ICL =1,IST
0305          I1 =(ICL-1)*12 +1
0306          I2 =I1 +11
0307          WRITE(5,117)
0308          IF(IPT.EQ.1) WRITE(5,1117)
0309          IF(IPT.EQ.2) WRITE(5,1127)
0310          WRITE(5,113)(IPP(J),J =I1,I2)
0311          DO 325 ILN =LINES,1,-1
0312          RB =(ILN-1)*12 +1
0313          CALL DPR(NEF,RB,H0,8,*256,)
0314          DO 340 IP =I1,I2
0315          NTPP =NTP0(2*IP)
0316          IF(NTPP .NE. 1) GO TO 345
0317          BUFF(IP) =0.
0318          GO TO 340
0319  345     IF(K0(IP) .LE. 0.) GO TO 350
0320          THICK =H0(IP) -BTM0(IP)
0321          IF(H0(IP) .GT. TOP0(IP)) THICK =TOP0(IP) -BTM0(IP)
0322          BUFF(IP) =THICK*K0(IP)*CNVT
0323          GO TO 340
0324  350     THICK =ABS(BTM0(IP))
0325          IF(THICK .EQ. 0.) THICK =1.
0326          BUFF(IP) =THICK*ABS(K0(IP))*CNVT
0327  340     CONTINUE
0328          WRITE(5,109) ILN,(BUFF(J),J =I1,I2)
0329  325     CONTINUE
0330          WRITE(5,107)
0331  330     CONTINUE
0332          CLOSE(UNIT=5)
0333          RETURN
0334  100     FORMAT(1X,'PRINT ?')
0335  101     FORMAT(2I6)
0336  102     FORMAT(9X,'N',9X,'M',9X,'LINES',5X,'COLL',6X,'NIT',7X,'ITCT'
              1,6X,'CVLIM'//)
0337  103     FORMAT(1X,6I10,F10.6)
0338  104     FORMAT(1X,'PRINT MATRIX ?')
0339  105     FORMAT(1X,'START LINE, STOP LINE')
0340  106     FORMAT(1X,I5,2X,12F10.3)
0341  107     FORMAT(1H1)
0342  108     FORMAT(1X, 'SOLUTION ,THICKNESS ,SOL -BIAS, STRESS'/
              1'  1          2          3          4')
0343  109     FORMAT(1X,I5,2X,12F10.0)
0344  110     FORMAT(1X,'Q -CAL , INT. AQUIFER TRANS. ,Q -MAG. ,Q -RATIO'
              1,' ,OCEAN LEAKAGE'/
              1'  1                2                3          4          5')
0345  111     FORMAT(8X,I7,11I10/)
0346  112     FORMAT(1X,'TRANSMISSIVITY MAP ?:(1) =FT**2/DAY ;(2) =GAL/DAY/FT')
0347  113     FORMAT(10X,I7,11I10/)
```

```
FORTRAN IV-PLUS V02-51D          16:26:33     30-MAY-79
PRNTR.FTN        /TR:BLOCKS/WR

0348     114     FORMAT(55X,'POTENTIAL SOLUTION (FT) BIAS= 'F10.3,/)
0349     115     FORMAT(50X,'THICKNESS (FT) (+=UNCONF. -=CONF.)'/)
0350     116     FORMAT(55X,'Q CHECK (CFD)'/)
0351     117     FORMAT(55X,'TRANSMISSIVITY'/)
0352     1117    FORMAT(1H+,71X,'FT**2/DAY')
0353     1127    FORMAT(1H+,71X,'GAL/DAY-FT')
0354     118     FORMAT(55X,'UNBIASED POTENTIAL (FT)'/)
0355     119     FORMAT(55X,'INTERAQUIFER TRANS. (CFD)'/)
0356     121     FORMAT(1X,'TOTAL FLOW (GPM) =',F10.2)
0357     122     FORMAT(55X,'AV. FLUX MAG. (CFD)'/)
0358     123     FORMAT(55X,'FLUX RATIO  (PERCENT)'/)
0359     124     FORMAT(1X,I5,2X,12F10.4)
0360     125     FORMAT(55X,'OCEAN TRANSFER (CFD)'/)
0361             END
```

```
FORTRAN IV-PLUS V02-510          16:30:08     30-MAY-79
NEWFLUX.FTN      /CK/TR:ALL/WR

      C ******  NEWFLUX.FTN ******
0001          COMMON/HDR/LNPT,NTT,NBPTC,NBFPTC,NSTREM,NPT,NSELEM,NELEM,NODMAX,
             #NONZER,NO,MATN,KTYPE,NMAX,NSTIF,NTIME,NSTORE,NDUM,TITLE(20),
             #DP1,DP2,DP3,DP4,DP5,DP6,DP7,DP89,DP10,DP11,DP12,DP13,DP14,DP15,
             #DP16,DP17,DP18
0002             COMMON/BF/NODBF(128),BIVF(128)
0003          WRITE(2,*)'ENTER FILE-U DISK UNIT'
0004          READ(2,*)IUN
0005        5 IF(IUN.NE.3)CALL ASNLUN(1,'DP',IUN,ISTAT)
0006             CALL DPFILE(1,'DESCRIP',25.,IPTR1)
0007             CALL DPR(IPTR1,0.,LNPT,92.)
0008          CALL DPFILE(1,'BOUNDCON',DP7,IPTR7)
0009             CALL DPR(IPTR7,26.,NODBF,128.)
0010             CALL DPR(IPTR7,27.,BIVF,256.)
0011             IFLAG = 0
0012             DO 100 I = 1 , NBFPTC
0013             WRITE(2,*)NODBF(I),BIVF(I)
0014             WRITE(2,*)'NEW FLUX (NO CHANGE = 0)'
0015             READ(2,50)BIV
0016     50      FORMAT(E10.3)
0017             IF(BIV.EQ.0.)GO TO 100
0018             BIVF(I) = BIV
0019             IFLAG = 1
0020    100      CONTINUE
0021             WRITE(2,*)' NEW NODES FOR FLUX(=0 NO ADDITION,=NEW NODES '
0022             READ(2,*)NYES
0023             IF(NYES.EQ.0)GO TO 200
0024             IFLAG = 1
0025             DO 150 I = 1, NYES
0026             NBFPTC = NBFPTC + 1
0027    150      READ(2,*)NODBF(NBFPTC),BIVF(NBFPTC)
0028    200      IF(IFLAG.EQ.0)GO TO 300
0029             CALL DPW(IPTR7,26.,NODBF,128.)
0030             CALL DPW(IPTR7,27.,BIVF,256.)
0031             IF(NYES.GT.0)CALL DPW(IPTR1,0.,LNPT,92.)
0032    300      STOP
0033             END
```

```
                                                  .TITLE RTRFAC
                                                  .GLOBL SHFTUP,SFTDWN,SEGFWD,SEGBWD,SEGFAC,S$SQRT
000000                                            .PSECT BKA,RW,D,OVR,GBL
000000  000000                    NE:             .WORD
000002  000000                    NEF:            .WORD
000004  000000                    M:              .WORD
000006  000000                    NTOT:           .WORD
000010  000000                    NBF:            .WORD
                                  ;
000000                                            .PSECT BUFF,RW,D,OVR,GBL
000000                            A:              .BLKW   17920.
                                  ;
000000                                            .PSECT PROGM,RW,I,CON,LCL
000000  000000                    NSOLV:          .WORD
000002  000000                    NSTRT:          .WORD
000004  000000                    NREL:           .WORD
000006  000000                    JL0:            .WORD
000010  000000                    JLN:            .WORD
000012  000000                    IMAX:           .WORD
000014  000000                    JLP:            .WORD
000016  000000                    I:              .WORD
                                  ;
        000000                    F0=R0
        000001                    F1=R1
        000002                    F2=R2
        000003                    F3=R3
        000004                    F4=R4
                                  ;
000020  017567  000002  177754    SEGFAC: MOV     @2(R5),NSTRT
000026  017500  000004                    MOV     @4(R5),R0
000032  017567  000006  177740            MOV     @6(R5),NSOLV
000040  010067  177740                    MOV     R0,NREL
000044  005300                            DEC     R0
000046  010067  177736                    MOV     R0,JLN
000052  170001                            SETF
000054  027527  000004  000001            CMP     @4(R5),#1
000062  003023                            BGT     SN6
000064  172467  000000*                   LDF     A,F0
000070  004767  000000G                   JSR     PC,SSSQRT
000074  174067  000000*                   STF     F0,A
000100  016700  000004*                   MOV     M,R0
000104  012701  000004*                   MOV     #A+4,R1
000110  172467  000000*                   LDF     A,F0
000114  172511                    LP5:    LDF     (R1),F1
000116  174500                            DIVF    F0,F1
000120  174121                            STF     F1,(R1)+
000122  077004                            SOB     R0,LP5
000124  012767  000001  177656            MOV     #1,JLN
000132  005267  177652            SN6:    INC     JLN
000136  170402                            CLRF    F2
000140  016700  177644                    MOV     JLN,R0
000144  166700  000004*                   SUB     M,R0
000150  020027  000001                    CMP     R0,#1
000154  002002                            BGE     1$
000156  012700  000001                    MOV     #1,R0
000162  016701  177622            1$:     MOV     JLN,R1
000166  160001                            SUB     R0,R1
```

```
000170  016702  177614                  MOV     JLN,R2
000174  160002                          SUB     R0,R2
000176  006302                          ASL     R2
000200  006302                          ASL     R2
000202  005300                          DEC     R0
000204  072027  000010                  ASH     #8.,R0
000210  062700  000000'                 ADD     #A,R0
000214  060200                          ADD     R2,R0           ;STARTING ADDRESS
000216  172510                  LP10:   LDF     (R0),F1
000220  171110                          MULF    (R0),F1
000222  172201                          ADDF    F1,F2
000224  062700  000374                  ADD     #252.,R0
000230  077106                          SOB     R1,LP10
000232  172410                          LDF     (R0),F0
000234  173002                          SUBF    F2,F0
000236  010046                          MOV     R0,-(SP)
000240  004767  000000G                 JSR     PC,$SSQRT
000244  012600                          MOV     (SP)+,R0
000246  174010                          STF     F0,(R0)
000250  016700  177526                  MOV     NSTRT,R0
000254  066700  177530                  ADD     JLN,R0
000260  005300                          DEC     R0
000262  020067  000006'                 CMP     R0,NTOT
000266  002401                          BLT     1$
000270  000207                          RTS     PC
000272  016701  000006'         1$:     MOV     NTOT,R1
000276  160301                          SUB     R0,R1
000300  005201                          INC     R1              ;NTOT =(NSTRT +JLN -1) +1
000302  016702  000004'                 MOV     M,R2
000306  005202                          INC     R2              ;M+1
000310  020201                          CMP     R2,R1
000312  003401                          BLE     2$
000314  010102                          MOV     R1,R2
000316  005302                  2S:     DEC     R2
000320  010205                          MOV     R2,R5           ;CTR
000322  012767  000002  177466          MOV     #2,I
000330  170400                  LP25:   CLRF    F0
000332  016700  000004'                 MOV     M,R0
000336  005200                          INC     R0
000340  026700  177452                  CMP     I,R0
000344  002067                          BGE     SN30
000346  016700  000004'                 MOV     M,R0
000352  005200                          INC     R0
000354  166700  177436                  SUB     I,R0
000360  016701  177424                  MOV     JLN,R1
000364  162701  000001                  SUB     #1,R1
000370  020001                          CMP     R0,R1
000372  003401                          BLE     1$
000374  010100                          MOV     R1,R0           ;K ALSO CTR
000376  016701  177406          1S:     MOV     JLN,R1
000402  160001                          SUB     R0,R1           ;J1
000404  005301                          DEC     R1
000406  072127  000010                  ASH     #8.,R1
000412  062701  177774'                 ADD     #A-4,R1         ;LINE ADDRESS
000416  016702  177374                  MOV     I,R2
000422  060002                          ADD     R0,R2
000424  006302                          ASL     R2
```

```
115 000426  006302                          ASL     R2
116 000430  060102                          ADD     R1,R2           ;K+I
117 000432  010003                          MOV     R0,R3
118 000434  005203                          INC     R3
119 000436  006303                          ASL     R3
120 000440  006303                          ASL     R3
121 000442  060301                          ADD     R3,R1           ;K +1
122 000444  172511                  LP15:   LDF     (R1),F1
123 000446  171112                          MULF    (R2),F1
124 000450  172001                          ADDF    F1,F0
125 000452  062701  000374                  ADD     #252.,R1
126 000456  062702  000374                  ADD     #252.,R2
127 000462  077010                          SOB     R0,LP15
128 000464  172512                  SN16:   LDF     (R2),F1
129 000466  173100                          SUBF    F0,F1
130 000470  174511                          DIVF    (R1),F1
131 000472  174112                          STF     F1,(R2)
132 000474  005267  177316                  INC     I
133 000500  077565                          SOB     R5,LP25
134 000502  016700  177276                  MOV     NREL,R0
135 000506  066700  177266                  ADD     NSOLV,R0
136 000512  005300                          DEC     R0
137 000514  020067  177270                  CMP     R0,JLN
138 000522  001204                          BNE     SN6
139 000522  000207                          RTS     PC
140 000524  016701  177260          SN30:   MOV     JLN,R1
141 000530  005301                          DEC     R1
142 000532  072127  000010                  ASH     #8.,R1
143 000536  062701  000000*                 ADD     #A,R1
144 000542  016702  000000*                 MOV     M,R2
145 000546  006302                          ASL     R2
146 000550  006302                          ASL     R2
147 000552  060102                          ADD     R1,R2
148 000554  000743                          BR      SN16
149                                 ;
150                                 ;
151 000556  017567  000002  177216  SEGFWD: MOV     #2(R5),NSTRT
152 000564  017567  000004  177206          MOV     #4(R5),NSOLV
153 000572  170001                          SETF
154 000574  012701  000000*                 MOV     #A,R1
155 000600  012767  000001  177202          MOV     #1,JLN
156 000606  010103                  SGLP:   MOV     R1,R3
157 000610  062703  000374                  ADD     #252.,R3
158 000614  010102                          MOV     R1,R2
159 000616  062702  000004                  ADD     #4,R2
160 000622  172413                          LDF     (R3),F0
161 000624  174411                          DIVF    (R1),F0
162 000626  174013                          STF     F0,(R3)
163 000630  016704  177154                  MOV     JLN,R4
164 000634  066704  177142                  ADD     NSTRT,R4
165 000640  005304                          DEC     R4
166 000642  010467  177140                  MOV     R4,JL0
167 000646  020467  000006*                 CMP     R4,NTOT
168 000652  002401                          BLT     1$
169 000654  000207                          RTS     PC
170 000656  016700  000006*         1$:     MOV     NTOT,R0
171 000662  166700  177120                  SUB     JL0,R0
```

```
000666  005200                          INC     R0
000670  016704  000004*                 MOV     M,R4
000674  005204                          INC     R4
000676  020004                          CMP     R0,R4
000700  003003                          BGT     2$
000702  010067  177104                  MOV     R0,IMAX
000706  000402                          BR      3$
000710  010467  177076          2$:     MOV     R4,IMAX
000714  016700  177072          3$:     MOV     IMAX,R0
000720  005300                          DEC     R0
000722  010304                          MOV     R3,R4
000724  062704  000400                  ADD     #256.,R4
000730  172413                          LDF     (R3),F0
000732  172514                  ILP:    LDF     (R4),F1
000734  172622                          LDF     (R2)+,F2
000736  171200                          MULF    F0,F2
000740  173102                          SUBF    F2,F1
000742  174114                          STF     F1,(R4)
000744  062704  000400                  ADD     #256.,R4
000750  077010                          SOB     R0,ILP
000752  005267  177032                  INC     JLN
000756  026767  177026  177014          CMP     JLN,NSOLV
000764  003401                          BLE     1$
000766  000207                          RTS     PC
000770  062701  000400          1$:     ADD     #256.,R1
000774  000704                          BR      SGLP
                                ;
                                ;
000776  017567  000002  176776  SEGBWD: MOV     #2(R5),NSTRT
001004  017567  000004  176772          MOV     #4(R5),NREL
001012  017567  000006  176760          MOV     #6(R5),NSOLV
001020  170001                          SETF
001022  016700  176756                  MOV     NREL,R0
001026  010067  176756                  MOV     R0,JLN
001032  166700  176742                  SUB     NSOLV,R0
001036  005200                          INC     R0
001040  010067  176742                  MOV     R0,JL0
001044  016700  176734                  MOV     NREL,R0
001050  005300                          DEC     R0
001052  072027  000010                  ASH     #8.,R0
001056  062700  000000*                 ADD     #A,R0
001062  010001                  MNLP:   MOV     R0,R1
001064  010002                          MOV     R0,R2
001066  062702  000374                  ADD     #252.,R2
001072  172412                          LDF     (R2),F0
001074  172510                          LDF     (R0),F1
001076  174401                          DIVF    F1,F0
001100  174012                          STF     F0,(R2)
001102  016703  176674                  MOV     NSTRT,R3
001106  166703  176672                  SUB     NREL,R3
001112  066703  176672                  ADD     JLN,R3
001116  010367  176672                  MOV     R3,JLP
001122  020327  000001                  CMP     R3,#1
001126  001001                          BNE     1$
001130  000207                          RTS     PC
001132  005303                  1$:     DEC     R3
001134  020367  000004*                 CMP     R3,M
```

```
001140  003003                                  BGT     2$
001142  010367  176644                          MOV     R3,IMAX
001146  000403                                  BR      3$
001150  016767  000004' 176634  2$:             MOV     M,IMAX
001156  016704  176630          3$:             MOV     IMAX,R4
001162  010001                                  MOV     R0,R1
001164  010203                                  MOV     R2,R3
001166  172412                                  LDF     (R2),F0
001170  162701  000374          INLP:           SUB     #252.,R1
001174  162703  000400                          SUB     #256.,R3
001200  172511                                  LDF     (R1),F1
001202  171100                                  MULF    F0,F1
001204  172613                                  LDF     (R3),F2
001206  173201                                  SUBF    F1,F2
001210  174213                                  STF     F2,(R3)
001212  077412                                  SOB     R4,INLP
001214  162700  000400                          SUB     #256.,R0
001220  005367  176564                          DEC     JLN
001224  026767  176560  176534                  CMP     JLN,JL0
001232  002313                                  BGE     MNLP
001234  000207                                  RTS     PC
                                        ;
                                        ;
001236  017567  000002  176534  SHFTUP:         MOV     @2(R5),NSOLV
001244  170001                                  SETF
001246  016700  000010'                         MOV     NBF,R0
001252  166700  176522                          SUB     NSOLV,R0
001256  072027  000007                          ASH     #7,R0
001262  012701  000000'                         MOV     #A,R1
001266  010102                                  MOV     R1,R2
001270  016703  176504                          MOV     NSOLV,R3
001274  072327  000010                          ASH     #8.,R3
001300  060302                                  ADD     R3,R2
001302  012221                          SFTLP:  MOV     (R2)+,(R1)+
001304  077002                                  SOB     R0,SFTLP
001306  000207                                  RTS     PC
                                        ;
                                        ;
001310  017567  000002  176462  SFTDWN:         MOV     @2(R5),NSOLV
001316  170001                                  SETF
001320  016700  000010'                         MOV     NBF,R0
001324  166700  176450                          SUB     NSOLV,R0
001330  072027  000007                          ASH     #7,R0
001334  012701  000000'                         MOV     #A,R1
001340  016702  000010'                         MOV     NBF,R2
001344  072227  000010                          ASH     #8.,R2
001350  060201                                  ADD     R2,R1
001352  010102                                  MOV     R1,R2
001354  016703  176420                          MOV     NSOLV,R3
001360  072327  000010                          ASH     #8.,R3
001364  160302                                  SUB     R3,R2
001366  014241                          DWNLP:  MOV     -(R2),-(R1)
001372  077002                                  SOB     R0,DWNLP
001372  000207                                  RTS     PC
        000001                                  .END
```

```
RTRFAC  MACRO D1013  30-MAY-79 16:30
SYMBOL TABLE

A       000000R     003 ILP     000732R     004 LP25    000330R     004 NSOLV   000000R     004 SFTLP   001302R     004
DWNLP   001366R     004 IMAX    000012R     004 LP5     000114R     004 NSTRT   000002R     004 SGLP    000606R     004
F0    =%000000          INLP    001170R     004 M       000004R     002 NTOT    000006R     002 SHFTUP  001236RG    004
F1    =%000001          JLN     000010R     004 MNLP    001062R     004 SEGBWD  000776RG    004 SN16    000464R     004
F2    =%000002          JLP     000014R     004 NBF     000010R     002 SEGFAC  000020RG    004 SN30    000524R     004
F3    =%000003          JL0     000006R     004 NE      000000R     002 SEGFWD  000556RG    004 SN6     000132R     004
F4    =%000004          LP10    000216R     004 NEF     000002R     002 SFTDWN  001310RG    004 $$SQRT= ****** G
I       000016R     004 LP15    000444R     004 NREL    000004R     004

. ABS.  000000      000
        000000      001
BKA     000012      002
BUFF    106000      003
PROGM   001374      004
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  297 WORDS  ( 2 PAGES)
DYNAMIC MEMORY:  11935 WORDS  ( 45 PAGES)
ELAPSED TIME:  00:00:20
.LP:=DB0:RTRFAC
```

```
DB0:VTTSS3=VTTSS3,ODL/MP
ASG=TI:6,LP:5,DB0:2
/
```

```
A:      .FCTR   VTTSS3
B:      .FCTR   [1,1]BNWLIB/LB
C:      .FCTR   READ
D:      .FCTR   VTTGN3
E:      .FCTR   MXSOLV
F:      .FCTR   PRNTR
G:      .FCTR   MXSING
        .ROOT   A-B-*(C,D,E-G,F)
        .END
```

```
        C       NAME:   VTTSS3
        C       PROGRAM: THIS PROGRAM USES NEWTONS METHOD ON THE UNSYMMETRIC
        C                MULTI AQUIFER VTT STEADY STATE EQUATION
        C
0001            BYTE STAT
0002            COMMON /BKA/ NE,NEF,M,N,NBF
0003            COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0004            WRITE(6,*)' NEWTON VERSION OF S.S.VTT '
0005            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0006            READ(6,*) IUN
0007            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0008            WRITE(6,106)
0009            READ(6,101) IFLD
0010            IF(IFLD.EQ.0) CALL OPFILE(2,'VTTFIELDS',2401.,NEF)
0011            IF(IFLD.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,NEF)
0012            CALL OPFILE(2,'VTTMATRIX',5000.,NE)
0013            NBF =80
0014            CALL OPR(NEF,0.,NN,8.,STAT)
0015            M =MM
0016            N =NN
0017            WRITE(6,104)
0018            READ(6,101) IG
0019            IF(IG .EQ. 0) GO TO 443
0020            WRITE(6,100)
0021            READ(6,101) M,N,ICOL,LINES,NIT,ITCT
0022    443     CALL PRNTR(0,2,0)
0023    444     WRITE(6,*)' ENTER THE NUMBER OF ITTERATIONS'
0024            READ(6,*,END=445)NIT
0025    20      DO 10 IT =1,NIT
0026            IF(IT.NE.1) DTOSS=SECNDS(TOSS)/60.
0027            ITOSS=IT-1
0028            IF(IT.NE.1) WRITE(6,107) DTOSS,ITOSS
0029    107     FORMAT(' TIME=',F10.2,' MIN. FOR ITERATION=',I5)
0030            WRITE(6,102) IT
0031            TOSS=SECNDS(0.)
0032            CALL VTTGN3
0033            CALL MXSOLV
0034            CALL READ
0035            ITCT =ITCT +1
0036    10      CONTINUE
0037            DTOSS=SECNDS(TOSS)/60.
0038            WRITE(6,107) DTOSS,NIT
0039            WRITE(6,105) NIT
0040            CALL PRNTR(0,2,0)
0041            WRITE(6,*)' ARE YOU FINISHED?   0=NO, 1=YES '
0042            READ(6,101)IANS
0043            IF(IANS.EQ.0) GO TO 444
0044    445     STOP
0045    100     FORMAT(1X,'INPUT M,N,ICOL,LINES,NIT,ITCT:6I5')
0046    101     FORMAT(6I6)
0047    104     FORMAT(1X,' TTY HEADER INPUT ?')
0048    102     FORMAT(1X,'START OF ITERATION',I3)
0049    105     FORMAT(1X,'FINISHED :',I4,2X,'ITERATIONS')
0050    106     FORMAT(1X,'LARGE OR SMALL FIELD ? :0=LG,1=SM')
0051            END
```

```
0001            SUBROUTINE READ
0002            BYTE STAT
0003            COMMON /BKA/ NE,NEF,M,N,N8F
0004            COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005            DIMENSION RBUF(128,63),BUFF(128,8),H(128),NNO(256),BTM(128)
                1,HYDC(128)
0006            EQUIVALENCE (BUFF(1,1),H),(BUFF(1,6),NNO),(BUFF(1,2),BTM)
                1,(BUFF(1,3),HYDC)
0007            RMX =0.
0008            DO 20 ILN =1,LINES
0009            RB =(ILN-1)*12 +1
0010            CALL DPR(NEF,RB,BUFF,8,*256,)
0011            NODE =NNO(1)
0012            IF(NODE .GT. 63) NODE =63
0013            IF(NODE .EQ. 0) GO TO 20
0014            CALL DPR(NE,RMX,RBUF,NODE*256,)
0015            DO 5 IT =1,NODE
0016            INDX =NNO(IT+1)
0017            H(INDX) =RBUF(2*M+2,IT)
0018     5      CONTINUE
0019            IND =NNO(2)
0020            IF(HYDC(IND) .LT. 0.) GO TO 1000
0021            DO 10 I =1,NODE
0022            IP =NNO(I+1)
0023            H0 =H(IP)
0024            BT0 =BTM(IP)
0025            IF(H0 .GT. BT0+.001) GO TO 10
0026            H(IP)=BT0+.001
0027            WRITE(5,100) ILN,IP,H(IP),BT0,H0
0028     10     CONTINUE
0029     1000   CONTINUE
0030            CALL DPW(NEF,RB,BUFF,2048,)
0031            RMX =RMX +NODE
0032     20     CONTINUE
0033            CLOSE(UNIT=5)
0034            RETURN
0035     100    FORMAT(1X,'POTENTIAL PROBLEM AT LINE =',I4,'  COL=',I4,5X
                1,'PHE =',F10.2,5X,' BOTTOM =',F10.2,5X,'OLD PHE =',F10.2)
0036            END
```

```
        C       ***** VTTGNS *****
        C       PROGRAM: SUBROUTINE FOR GENERATING SYMMETRIC 3 AQUIFER
        C               CONFINED - UNCONFINED MATRIX
        C
0001            SUBROUTINE VTTGNS
0002            BYTE STAT
0003            COMMON /BKA/ NE,NEF,M,N,NBF
0004            COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005            DIMENSION TRBUF(64,80),ACOFFR(4,27)
0006            REAL H2(128),H0(128),H4(128),BTM2(128),BTM0(128),BTM4(128)
                1,K2(128),K0(128),K4(128),Q0(128),TRN0(128),KEF
                2,TOP2(128),TOP0(128),TOP4(128)
0007            INTEGER LNBF(256,24),NTP2(256),NTP0(256),NTP4(256),NNO2(256)
                1,NNO0(256),NNO4(256),INDX3(4,3),IHDD(12)
0008            EQUIVALENCE (LNBF(1,1),H2),(LNBF(1,9),H0),(LNBF(1,17),H4)
                1,(LNBF(1,2),BTM2),(LNBF(1,10),BTM0),(LNBF(1,18),BTM4)
                2,(LNBF(1,3),K2),(LNBF(1,11),K0),(LNBF(1,19),K4)
                3,(LNBF(1,12),Q0),(LNBF(1,5),NTP2),(LNBF(1,13),NTP0)
                4,(LNBF(1,21),NTP4),(LNBF(1,6),NNO2),(LNBF(1,14),NNO0)
                5,(LNBF(1,22),NNO4),(LNBF(1,16),TRN0),(IHDD(11),NAQF)
                6,(LNBF(1,7),TOP2),(LNBF(1,15),TOP0),(LNBF(1,23),TOP4)
0009            DIMENSION H(7),BTM(5),K(5),NTP(6),NNO(7),CN(7),TN(5),AA(4)
0010            DIMENSION TAV(4),TOP(5)
0011            REAL K
0012            DATA ACOFFR/1.,1.,1.,1. ,0.,0.,0.,0. ,0.,0.,0.,0.
                1, .5,1.,.5,0. ,0.,1.,1.,0. ,0.,.5,1.,.5
                2 ,0.,0.,1.,1., .5,0.,.5,1. ,1.,0.,0.,1.
                3 ,1.,.5,0.,.5 ,1.,1.,0.,0. ,0.,1.,.5,0.
                5 ,0.,.5,1.,0. ,0.,0.,1.,.5 ,0.,0.,.5,1.
                6 ,.5,0.,0.,1., 1.,0.,0.,.5 ,1.,.5,0.,0.
                7 ,.5,1.,0.,0. ,1.,0.,.5,1. ,1.,.5,0.,1.
                8 ,1.,1.,0.,.5 ,1.,1.,.5,0. ,.5,1.,1.,0.
                9 ,0.,1.,1.,.5 ,0.,.5,1.,1., .5,0.,1.,1./
0013            DATA INDX3/1,2,0,2,  -1,1,-1,0,  -2,-1,0,-1/
        C
        C
0014            CALL DPR(NEF,0.,IHDD(1),12.)
0015            ICOL3 =ICOL/NAQF
0016            IWBF =80
0017            IWCT =0
0018            H(6) =0.
0019            H(7) =0.
0020            TR5 =0.
0021            TR6 =0.
0022            NTP(5) =1
0023            NTP(6) =1
0024            ITR =1
0025    1       DO 5 I =1,M
0026            DO 5 J =1,8
0027    5       LNBF(I,J) =0.
0028            CALL DPR(NEF,1.,H0,8.*256.)
0029            CALL DPR(NEF,13.,H4,8.*256.)
0030            DO 70 ILN =1,LINES
0031            NDPTS =NNO0(1)
0032            IF(NDPTS .EQ. 0) GO TO 60
0033            DO 60 INODE =1,NDPTS
```

```
0034            IP =NNO0(INODE+1)
0035            IAQF =(IP-1)/ICOL3 +1
0036            IH5 =INDX3(1,IAQF)*ICOL3 +IP
0037            IH6 =INDX3(2,IAQF)*ICOL3 +IP
0038            IT5 =INDX3(3,IAQF)*ICOL3 +IP
0039            IT6 =INDX3(4,IAQF)*ICOL3 +IP
0040            H(1) =H0(IP)
0041            BTM(1) =BTM0(IP)
0042            TOP(1) =TOP0(IP)
0043            K(1) =K0(IP)
0044            NTP(1) =NTP0(2*IP+2)
0045            NNO(1) =NTP0(2*IP-1)
0046            H(2) =H0(IP+1)
0047            BTM(2) =BTM0(IP+1)
0048            TOP(2) =TOP0(IP+1)
0049            K(2) =K0(IP+1)
0050            NTP(2) =NTP2(2*IP)
0051            NNO(2) =NTP0(2*IP+1)
0052            H(3) =H2(IP)
0053            BTM(3) =BTM2(IP)
0054            TOP(3) =TOP2(IP)
0055            K(3) =K2(IP)
0056            NNO(3) =NTP2(2*IP-1)
0057            IF(IP .EQ. 1) GO TO 11
0058            H(4) =H0(IP-1)
0059            BTM(4) =BTM0(IP-1)
0060            TOP(4) =TOP0(IP-1)
0061            K(4) =K0(IP-1)
0062            NTP(3) =NTP0(2*IP-2)
0063            NNO(4) =NTP0(2*IP-3)
0064     11     H(5) =H4(IP)
0065            BTM(5) =BTM4(IP)
0066            TOP(5) =TOP4(IP)
0067            K(5) =K4(IP)
0068            NTP(4) =NTP4(2*IP)
0069            NNO(5) =NTP4(2*IP-1)
0070            NDTYP =NTP0(2*IP)
0071            GO TO (201,202,203),NAQF
0072     203    H(7) =H0(IH6)
0073            NTP(6) =NTP0(2*IH6)
0074            NNO(7) =NTP0(2*IH6-1)
0075            TR6 =TRN0(IT6)
0076     202    H(6) =H0(IH5)
0077            NTP(5) =NTP0(2*IH5)
0078            NNO(6) =NTP0(2*IH5-1)
0079            TR5 =TRN0(IT5)
0080     201    IF(NDTYP .GT. 2) NDTYP =NDTYP-5
0081            DO 10 I =1,4
0082     10     AA(I) =ACOFFR(I,NDTYP+1)
         C
         C                          COMPUTE COFF.
         C
0083            DO 15 I =1,5
0084            BB =BTM(I)
0085            IF(K(I) .GE. 0) GO TO 12
0086            IF(BB .EQ. 0.) BB =1.
```

```
0087            TN(I) =ABS(K(I))*BB
0088            GO TO 15
0089     12     THICK =H(I) -BB
0090            IF(H(I) .GT. TOP(I)) THICK =TOP(I) -BB
0091            TN(I) =K(I)*THICK
0092     15     CONTINUE
0093            DO 20 I =1,4
0094     20     TAV(I) =(TN(I) +TN(I+1))/2.
0095            DO 25 I =1,4
0096            CN(I+1) =AA(I)*TAV(I)
0097            IF(NTP(I) .EQ. 2) CN(I+1) =0.
0098     25     CONTINUE
0099            IF(NTP(5) .EQ. 1) TR5 =0.
0100            IF(NTP(6) .EQ. 1) TR6 =0.
0101            CN(6) =TR5
0102            CN(7) =TR6
0103            IF(NTP(5) .EQ. 2) CN(6) =0.
0104            IF(NTP(6) .EQ. 2) CN(7) =0.
0105            SM3 =0.
0106            DO 30 I =1,4
0107            SM3 =SM3 +AA(I)*TAV(I)
0108     30     CONTINUE
0109            CN(1) =-SM3 -TR5 -TR6
0110            F0 =0.
0111            DO 35 I =1,4
0112            F0 =F0 +AA(I)*TAV(I)*(H(I+1) -H(I))
0113     35     CONTINUE
0114            F0 =F0 + Q0(IP) +TR5*(H(6) -H(1)) +TR6*(H(7) -H(1))
0115            BTRM =0.
0116            DO 40 I =1,4+NAQF
0117     40     BTRM =BTRM +CN(I)*H(I)
0118            BTRM =BTRM -F0
0119     41     CONTINUE
        C
        C                       STORE COFF IN MARTIX
        C
0120            DO 45 I =1,M+1
0121     45     TRBUF(I,ITR) =0.
0122            TRBUF(64,ITR) =0.
0123            DO 50 I =1,4+NAQF
0124            IF(CN(I) .EQ. 0) GO TO 50
0125            INDX =1 +NNO(I) -NNO(1)
0126            IF(INDX .GT. M+1) GO TO 50
0127            IF(INDX .LT. 1) GO TO 50
0128            TRBUF(INDX,ITR) = -CN(I)
0129     50     CONTINUE
0130            TRBUF(64,ITR) = -BTRM
0131            IF(NNO(1) .EQ. N) GO TO 55
0132            IF(ITR .EQ. IWBF) GO TO 55
0133            ITR =ITR +1
0134            GO TO 60
0135     55     WRLC =IWCT/2
0136            CALL DPW(NE,WRLC,TRBUF,ITR*128.)
0137            ITR =1
0138            IWCT =IWCT +IWBF
0139     60     CONTINUE
```

```
        C
        C                          SHIFT AND READ
        C
0140            DO 65 I =1,2*ICOL
0141            DO 65 J =1,16
0142            LNBF(I,J) =LNBF(I,J+8)
0143    65      CONTINUE
0144            RRD =(ILN +1)*12 +1
0145            IF(ILN +2 .GT. LINES) GO TO 70
0146            CALL DPR(NEF,RRD,H4,8.*256.)
0147    70      CONTINUE
0148            RETURN
0149            END
```

```
0001            SUBROUTINE MXSOLV
0002            BYTE STAT
0003            COMMON /BKA/ NE,NEQ,M,N,NBF
0004            COMMON /MATRX/ A(128,80),NSTRT
0005            RTF =NBF*256.
0006            CALL DPR(NE,0.,A,RTF)
0007            NSOLV =NBF -M
0008            KSEG =0
0009     1      KSEG =KSEG +1
0010            RLW =(KSEG-1)*NSOLV
0011            NSTRT =RLW +1
0012            CALL FWARD(1,NSOLV)
0013            CALL DPW(NE,RLW,A,NSOLV*256.)
0014            CALL SHIFT1(NSOLV)
0015            RLR =RLW+NBF
0016            NLST =RLR +NSOLV
0017            IF(NLST .GE. N) GO TO 5
0018            CALL DPR(NE,RLR,A(1,M+1),NSOLV*256.)
0019            GO TO 1
0020     5      NOVER =NLST -N
0021            NRED =NSOLV -NOVER
0022            NSTRT =RLW +NSOLV +1
0023            WBK = NSTRT -1
0024            NSOLV =NBF -NOVER
0025            CALL DPR(NE,RLR,A(1,M+1),NRED*256.)
0026            CALL FWARD(1,NSOLV)
0027            CALL DPW(NE,WBK,A,NSOLV*256.)
0028    10      KSEG =0
0029            NSOLV =NBF -M
0030            RL =N -NBF
0031            CALL DPR(NE,RL,A,RTF)
0032    11      KSEG =KSEG +1
0033            RLW =N -KSEG*(NSOLV -1)
0034            NSTRT =RLW +NSOLV -1
0035            CALL BKWARD(NBF,NSOLV)
0036            CALL DPW(NE,RLW,A(1,M+2),(NSOLV-1)*256.)
0037            CALL SHIFT2(NSOLV)
0038            RLR =RLW -NBF
0039            IF(RLR .LE. 0.) GO TO 15
0040            CALL DPR(NE,RLR,A,(NSOLV-1)*256.)
0041            GO TO 11
0042    15      NSOLV =NBF +RLR
0043            NSTRT =NBF
0044            RED =NBF -M +RLR
0045            JA =-RLR +1
0046            CALL DPR(NE,0.,A(1,JA),RED*256.)
0047            CALL BKWARD(NBF,NSOLV)
0048            CALL DPW(NE,0.,A(1,JA),NSOLV*256.)
0049    20      CONTINUE
0050            RETURN
0051            END
```

```
C          ****** SUBROUTINE PRNTR ******
0001            SUBROUTINE PRNTR(LWRAQ,IMETH,IOLD)
0002            BYTE STAT,FLG(2)
0003            COMMON /BKA/ NE,NEF,M,N,NBF
0004            COMMON /HEAD/ NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0005            DIMENSION BUFF(128),ACOFF(4,27),ACOFFR(4,27)
0006            REAL H2(128),H0(128),H4(128),BTM2(128),BTM0(128),BTM4(128)
                1,K2(128),K0(128),K4(128),Q0(128),TRN0(128),TOP2(128),BUFB(64,2)
                2,TOP0(128),TOP4(128),OCN0(128),LEAK,LEAKK
0007            INTEGER LNBF(256,27),NTP2(256),NTP0(256),NTP4(256),NNO2(256)
                1,NNO0(256),NNO4(256),IHDR(90),IPP(128),INDX3(4,3)
0008            EQUIVALENCE (LNBF(1,1),H2),(LNBF(1,2),BTM2),(LNBF(1,3),K2)
                1,(LNBF(1,5),NTP2),(LNBF(1,7),TOP2),(LNBF(1,10),H0)
                2,(LNBF(1,11),BTM0),(LNBF(1,12),K0),(LNBF(1,13),Q0)
                3,(LNBF(1,14),NTP0),(LNBF(1,15),NNO0),(LNBF(1,16),TOP0)
                4,(LNBF(1,17),TRN0),(LNBF(1,18),OCN0),(LNBF(1,19),H4)
                5,(LNBF(1,20),BTM4),(LNBF(1,21),K4),(LNBF(1,23),NTP4)
                6,(LNBF(1,25),TOP4),(BUFF,BUFB)
0009            EQUIVALENCE (IHDR(44),FLG),(IHDR(87),LEAK),(IHDR(89),HSEA)
                1,(IHDR(9),BIAS),(IHDR(11),NAQF)
0010            DIMENSION H(7),BTM(5),K(5),NTP(6),TN(5),AA(4),TOP(5),NNO(7)
0011            DIMENSION TAV(6),AREA(4)
0012            REAL K,KEF
0013            DATA ACOFFR/1.,1.,1.,1. ,0.,0.,0.,0. ,0.,0.,0.,0.
                1, .5,1.,.5,0. ,0.,1.,1.,0. ,0.,.5,1.,.5
                2 ,0.,0.,1.,1., .5,0.,.5,1. ,1.,0.,0.,1.
                3 ,1.,.5,0.,.5 ,1.,1.,0.,0. ,0.,1.,.5,0.
                5 ,0.,.5,1.,0. ,0.,0.,1.,.5 ,0.,0.,.5,1.
                6 ,.5,0.,0.,1., 1.,0.,0.,.5 ,1.,.5,0.,0.
                7 ,.5,1.,0.,0. ,1.,0.,.5,1. ,1.,.5,0.,1.
                8 ,1.,1.,0.,.5 ,1.,1.,.5,0. ,.5,1.,1.,0.
                9 ,0.,1.,1.,.5 ,0.,.5,1.,1., .5,0.,1.,1./
0014            DATA INDX3/1,2,0,2,  -1,1,-1,0,  -2,-1,0,-1/
0015            DATA AREA/1.,.5,.375,.625/
      C
      C
0016    1       WRITE(6,100)
0017            READ(6,101,END=1001) IP
0018            IF(IP .EQ. 0) RETURN
0019            IF(IP.NE.0) GO TO 1000
0020    1001    STOP
0021    1000    CONTINUE
0022            DO 2 I =1,128
0023    2       IPP(I) =I
0024            CALL DPR(NEF,0.,IHDR,90.)
0025            LEAKK=LEAK
0026            IF(FLG(2) .EQ. 0) LEAK =0.
      C         DEEPLV=BIASS
0027            ICOL3 =ICOL/NAQF
0028            H(6) =0.
0029            H(7) =0.
0030            TR5 =0.
0031            TR6 =0.
0032            NTP(5) =1
0033            NTP(6) =1
0034            WRITE(6,102)
```

```
0035              WRITE(5,103) NN,MM,LINES,ICOL,NIT,ITCT,CVLM
0036              WRITE(5,107)
        C
        C                          PRINT MATRIX
        C
0037              WRITE(6,104)
0038              READ(6,101) IP
0039              IF(IP .EQ. 0) GO TO 20
0040      3       WRITE(6,105)
0041              READ(6,101) IST,IND
0042              IF(IST .EQ. 0) GO TO 20
0043              IW =M+2
0044              IF(IMETH.EQ.2) IW=2*M+2
0045              INPT =IW/12
0046              IF(MOD(IW,12) .NE. 0) INPT =INPT +1
0047              DO 15 ICL =1,INPT
0048              I1 =(ICL-1)*12 +1
0049              I2 =I1 +11
0050              WRITE(5,111)(IPP(J),J =I1,I2)
0051              DO 10 ILL =IST,IND
0052              IF(IMETH.EQ.2) GO TO 1113
0053              RL =2.*((ILL-1)/2)
0054              CALL DPR(NE,RL/2.,BUFB,256.)
0055              ILA =MOD(ILL-1,2) +1
0056              BUFB(M+2,ILA) =BUFB(64,ILA)
0057              WRITE(5,109)ILL,(BUFB(J,ILA),J =I1,I2)
0058              GO TO 10
0059      1113    CONTINUE
0060              RL=ILL-1
0061              CALL DPR(NE,RL,BUFF,256.,STAT)
0062              WRITE(5,109)ILL,(BUFF(J),J=I1,I2)
0063      10      CONTINUE
0064              WRITE(5,107)
0065      15      CONTINUE
0066              CLOSE(UNIT=5)
0067              GO TO 3
        C
        C                          PRINT SOLUTION OR THICKNESS
        C
0068      20      WRITE(6,108)
0069              READ(6,101) IPI
0070              IF(IPI .EQ. 0) GO TO 40
0071              IST =ICOL/12
0072              IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0073              DO 30 ICL =1,IST
0074              I1 =(ICL-1)*12 +1
0075              I2 =I1 +11
0076              IF(IPI .EQ. 1) WRITE(5,114)BIAS
0077              IF(IPI .EQ. 2) WRITE(5,115)
0078              IF(IPI .EQ. 3) WRITE(5,118)
0079              IF(IPI.EQ.4) WRITE(5,126)
0080      126     FORMAT(55X,'STRESS (CFU)'/)
0081              WRITE(5,111)(IPP(J),J =I1,I2)
0082              QTOT=0.
0083              DO 25 ILNS =LINES,1,-1
0084              RL =(ILNS-1)*12 +1
```

```
0085              CALL DPR(NEF,RL,H0,8,*256.,STAT)
0086              IF(IPI.NE.4) GO TO 21333
0087              IF(ICL.NE.1) GO TO 21333
0088              DO 21332 IU=1,ICOL
0089     21332    QTOT=QTOT+H0(IU)
0090              QGPM=QTOT/192.5
0091     21333    CONTINUE
0092              GO TO (35,21,24,24),IPI
0093     21       DO 22 IP =I1,I2
0094              THICK =H0(IP) -BTM0(IP)
0095              IF(H0(IP) .GT. TOP0(IP)) THICK =BTM0(IP) -TOP0(IP)
0096              BUFF(IP) =THICK
0097     22       CONTINUE
0098              GO TO 27
0099     35       DO 36 IP =I1,I2
0100     36       BUFF(IP) =H0(IP)
0101              GO TO 27
0102     24       DO 26 IP =I1,I2
0103              OFF =BIAS
0104              IF(NTP0(2*IP) .EQ. 1) OFF =0.
0105              IF(IP .GT. ICOL) OFF =0.
0106              BUFF(IP) =H0(IP) -OFF
0107              IF(IPI.EQ.4) BUFF(IP)=Q0(IP)
0108     26       CONTINUE
0109     27       IF(IPI.NE.4)WRITE(5,106)ILNS,(BUFF(J),J =I1,I2)
0110              IF(IPI.EQ.4)WRITE(5,166)ILNS,(BUFF(J),J=I1,I2)
0111     166      FORMAT(1X,I5,2X,12F10.1)
0112     25       CONTINUE
0113              IF(IPI.EQ.4.AND.ICL.EQ.IST) WRITE(5,*) 'TOTAL STRESS GPM=',
                  1QGPM
0114              WRITE(5,107)
0115     30       CONTINUE
0116              CLOSE(UNIT=5)
0117              GO TO 20
         C
         C                          PRINT FLOW BALANCE
         C
0118     40       WRITE(6,110)
0119              READ(6,101) IPT
0120              IF(IPT .EQ. 0) GO TO 300
0121              IST =ICOL/12
0122              IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0123              QTOTAL =0.
0124              DO 50 ICL =1,IST
0125              I1 =(ICL-1)*12 +1
0126              I2 =I1 +11
0127              IF(IPT .EQ. 1) WRITE(5,116)
0128              IF(IPT .EQ. 2) WRITE(5,119)
0129              IF(IPT .EQ. 3) WRITE(5,122)
0130              IF(IPT .EQ. 4) WRITE(5,123)
0131              IF(IPT.EQ.5) WRITE(5,125)
0132              WRITE(5,113)(IPP(J),J =I1,I2)
0133              DO 74 ILN =LINES,2,-1
0134              RR =(ILN -2)*12 +1
0135              CALL DPR(NEF,RR,H2,9.*256.)
0136              CALL DPR(NEF,RR+12.,H0,9.*256.)
```

```
0137              CALL DPR(NEF,RB+24.,H4,9,*256.)
0138              DO 44 IZ =I1,I2
0139      44      BUFF(IZ) =0.
0140              IF(NNDØ(1) .EQ. 0) GO TO 60
0141              DO 60 IP =I1,I2
0142              IF(IP .GT. ICOL) GO TO 60
0143              F0 =0.
0144              NTPP =NTPØ(2*IP)
0145              IF(NTPP .EQ. 1 .OR. NTPP .EQ. 2) GO TO 60
0146              H(1) =HØ(IP)
0147              IAQF =(IP-1)/ICOL3 +1
0148              IH5 =INDX3(1,IAQF)*ICOL3 +IP
0149              IH6 =INDX3(2,IAQF)*ICOL3 +IP
0150              IT5 =INDX3(3,IAQF)*ICOL3 +IP
0151              IT6 =INDX3(4,IAQF)*ICOL3 +IP
0152              LEAK=0.
0153              CTRN=0.
0154              DEEP=0.
0155              IF(.NOT.(IAQF.EQ.1.AND.FLG(2).EQ.1))GO TO 1089
0156              CTRN=UCNØ(IP)
          C       DEEP=TRNØ(IP)-DEEPLV+HØ(IH5)
0157              DEEP=TRNØ(IP)+(HØ(IH5)-TRNØ(IH5))
0158              LEAK=LEAKK
0159      1089    CONTINUE
0160              IF(FLG(2).EQ.0) LEAK=0.
0161              IF(LWRAQ.EQ.0) LEAK=0.
0162              H(1) =HØ(IP)
0163              BTM(1) =BTMØ(IP)
0164              TOP(1) =TOPØ(IP)
0165              K(1) =KØ(IP)
0166              NTP(1) =NTPØ(2*IP+2)
0167              NNO(1) =NTPØ(2*IP-1)
0168              H(2) =HØ(IP+1)
0169              BTM(2) =BTMØ(IP+1)
0170              TOP(2) =TOPØ(IP+1)
0171              K(2) =KØ(IP+1)
0172              NTP(2) =NTP2(2*IP)
0173              NNO(2) =NTPØ(2*IP+1)
0174              H(3) =H2(IP)
0175              BTM(3) =BTM2(IP)
0176              TOP(3) =TOP2(IP)
0177              K(3) =K2(IP)
0178              NNO(3) =NTP2(2*IP-1)
0179              IF(IP .EQ. 1) GO TO 11
0180              H(4) =HØ(IP-1)
0181              BTM(4) =BTMØ(IP-1)
0182              TOP(4) =TOPØ(IP-1)
0183              K(4) =KØ(IP-1)
0184              NTP(3) =NTPØ(2*IP-2)
0185              NNO(4) =NTPØ(2*IP-3)
0186      11      H(5) =H4(IP)
0187              BTM(5) =BTM4(IP)
0188              TOP(5) =TOP4(IP)
0189              K(5) =K4(IP)
0190              NTP(4) =NTP4(2*IP)
0191              NNO(5) =NTP4(2*IP-1)
```

```
0192              NDTYP =NTP0(2*IP)
0193              ARF=AREA(NDTYP/8+1)
0194              IF(NDTYP.EQ.1) ARF=0.
0195              GO TO (201,202,203),NAQF
0196      203     H(7) =H0(IH6)
0197              NTP(6) =NTP0(2*IH6)
0198              NNO(7) =NTP0(2*IH6-1)
0199              TR6 =TRN0(IT6)
0200      202     H(6) =H0(IH5)
0201              NTP(5) =NTP0(2*IH5)
0202              NNO(6) =NTP0(2*IH5-1)
0203              TR5 =TRN0(IT5)
0204      201     IF(NDTYP .GT. 2) NDTYP =NDTYP-5
0205              IF(FLG(2).EQ.0) GO TO 2015
0206              H(7)=0.
0207              NTP(5)=1
0208              NTP(6)=1
0209              NNO(7)=0
0210              NNO(6)=0
0211              TR5=0.
0212              TR6=0.
0213              H(6)=0.
0214              IF(.NOT.(IAQF.EQ.1)) GO TO 2014
0215              IF(LWRAQ.EQ.0) GO TO 2015
        C         H(6)=H0(IH5)-DEEPLV+TRN0(IP)
0216              H(6)=TRN0(IP)+(H0(IH5)-TRN0(IH5))
0217              NNO(6)=NTP0(2*IH5-1)
0218              NTP(5)=NTP0(2*IH5)
0219              ARFF=AREA(NTP(5)/8+1)
0220              IF(NTP(5).EQ.1) ARFF=0.
0221              ARF=MIN(ARFF,ARF)
0222              TR5=LEAKK*ARF
0223              GO TO 2015
0224      2014    CONTINUE
0225              IF(.NOT.(IAQF.EQ.2)) GO TO 2015
0226              IF(LWRAN.EQ.0) GO TO 2015
        C         H(6)=H0(IH5)+DEEPLV-TRN0(IH5)
0227              H(6)=H0(IH5)+TRN0(IP)-TRN0(IH5)
0228              NNO(6)=NTP0(2*IH5-1)
0229              NTP(5)=NTP0(2*IH5)
0230              ARFF=AREA(NTP(5)/8+1)
0231              IF(NTP(5).EQ.1) ARFF=0.
0232              ARF=MIN(ARFF,ARF)
0233              TR5=LEAKK*ARF
0234      2015    CONTINUE
0235              DO 510 I =1,4
0236      510     AA(I) =ACOFFR(I,NDTYP+1)
0237              DEEPOF=0.
0238              TR8=0.
0239              IF(IOLD.NE.1) GO TO 3015
0240              POTDIF=0.
0241              CTRN=0.
0242              TR5=0.
0243              TR6=0.
0244              IF(IAWF.NE.1) GO TO 3015
0245              CTRN=GCN0(IP)
```

```
0246            IF(LWRAQ.EQ.0) GO TO 3015
0247            DEEPDF=TRN0(IP)-H(1)
0248            TR8=LEAKK*ARF
0249    3015    CONTINUE
        C
        C                       COMPUTE NET Q ,OR INT AQUIFER TRANSFER
        C
0250            DO 515 I =1,5
0251            BB =BTM(I)
0252            IF(K(I) .GE. 0) GO TO 512
0253            IF(BB .EQ. 0.) BB =1.
0254            TN(I) =ABS(K(I))*BB
0255            GO TO 515
0256    512     THICK =H(I) -BB
0257            IF(H(I) .GT. TOP(I)) THICK =TOP(I) -BB
0258            TN(I) =K(I)*THICK
0259    515     CONTINUE
0260            DO 520 I =1,4
0261    520     TAV(I) =(TN(I) +TN(I+1))/2.
0262            IF(NTP(5) .EQ. 1) TR5 =0.
0263            IF(NTP(6) .EQ. 1) TR6 =0.
0264            F0 =0.
0265            QMAG =0.
0266            DO 535 I =1,4
0267            FLUX = AA(I)*TAV(I)*(H(I+1) -H(I))
0268            F0 =F0 +FLUX
0269            QMAG =QMAG +ABS(FLUX)
0270    535     CONTINUE
0271            ADIV =0.
0272            DO 540 I =1,4
0273    540     ADIV =ADIV +AA(I)
0274            TRNF =TR5*(H(6) -H(1)) +  TR6*(H(7) -H(1))+TR8*DEEPDF
0275            OCNXF=CTRN*(HSEA-H(1))
0276            IF(ADIV .EQ. 0.) GO TO 60
0277            ABFLX =(QMAG )/ADIV
0278            IF(ABFLX .EQ. 0.) GO TO 60
0279            FLRAT =100.*ABS(F0/ABFLX)
0280            IF(IPT .EQ. 1) BUFF(IP) =-F0 -TRNF-OCNXF
0281            IF(IPT .EQ. 2) BUFF(IP) =TRNF
0282            IF(IPT .EQ. 3) BUFF(IP) =ABFLX
0283            IF(IPT .EQ. 4) BUFF(IP) =FLRAT
0284            IF(IPT.EQ.5) BUFF(IP)=OCNXF
0285            QTOTAL =QTOTAL +BUFF(IP)
0286    60      CONTINUE
0287            IF(IPT .NE. 4) WRITE(5,109) ILN,(BUFF(J),J =I1,I2)
0288            IF(IPT .EQ. 4) WRITE(5,124) ILN,(BUFF(J),J =I1,I2)
0289    70      CONTINUE
0290            WRITE(5,107)
0291    50      CONTINUE
0292            GPM =QTOTAL/192.5
0293            IF(IP1.NE.4)WRITE(5,121) GPM
0294            WRITE(5,107)
0295            CLOSE(UNIT=5)
0296            GO TO 40
        C
        C                       PRINT TRANSMISSIVITY
```

```
      C
0297  300   WRITE(6,112)
0298        READ(6,101) IPT
0299        IF(IPT .EQ. 0) RETURN
0300        CNVT =1.
0301        IF(IPT .EQ. 2) CNVT =7.481
0302        IST =ICOL/12
0303        IF(MOD(ICOL,12) .NE. 0) IST =IST +1
0304        DO 330 ICL =1,IST
0305        I1 =(ICL-1)*12 +1
0306        I2 =I1 +11
0307        WRITE(5,117)
0308        IF(IPT.EQ.1) WRITE(5,1117)
0309        IF(IPT.EQ.2) WRITE(5,1127)
0310        WRITE(5,113)(IPP(J),J =I1,I2)
0311        DO 325 ILN =LINES,1,-1
0312        RB =(ILN-1)*12 +1
0313        CALL DPH(NEF,RB,H0,8,*256.)
0314        DO 340 IP =I1,I2
0315        NTPP =NTP0(2*IP)
0316        IF(NTPP .NE. 1) GO TO 345
0317        BUFF(IP) =0.
0318        GO TO 340
0319  345   IF(K0(IP) .LE. 0.) GO TO 350
0320        THICK =H0(IP) -BTM0(IP)
0321        IF(H0(IP) .GT. TOP0(IP)) THICK =TOP0(IP) -BTM0(IP)
0322        BUFF(IP) =THICK*K0(IP)*CNVT
0323        GO TO 340
0324  350   THICK =ABS(BTM0(IP))
0325        IF(THICK .EQ. 0.) THICK =1.
0326        BUFF(IP) =THICK*ABS(K0(IP))*CNVT
0327  340   CONTINUE
0328        WRITE(5,109) ILN,(BUFF(J),J =I1,I2)
0329  325   CONTINUE
0330        WRITE(5,107)
0331  330   CONTINUE
0332        CLOSE(UNIT=5)
0333        RETURN
0334  100   FORMAT(1X,'PRINT ?')
0335  101   FORMAT(2I6)
0336  102   FORMAT(9X,'N',9X,'M',9X,'LINES',5X,'COLL',6X,'NIT',7X,'ITCT'
            1,6X,'CVLIM'//)
0337  103   FORMAT(1X,6I10,F10.6)
0338  104   FORMAT(1X,'PRINT MATRIX ?')
0339  105   FORMAT(1X,'START LINE, STOP LINE')
0340  106   FORMAT(1X,I5,2X,12F10.3)
0341  107   FORMAT(1H1)
0342  108   FORMAT(1X, 'SOLUTION ,THICKNESS ,SOL -BIAS, STRESS'/
            1'  1            2            3            4')
0343  109   FORMAT(1X,I5,2X,12F10.0)
0344  110   FORMAT(1X,'Q -CAL , INT. AQUIFER TRANS. ,Q -MAG. ,Q -RATIO'
            1,' ,OCEAN LEAKAGE'/
            1'  1                2                3            4            5')
0345  111   FORMAT(8X,I7,11I10/)
0346  112   FORMAT(1X,'TRANSMISSIVITY MAP ?:(1) =FT**2/DAY ?(2) =GAL/DAY/FT')
0347  113   FORMAT(10X,I7,11I10/)
```

```
0348    114     FORMAT(55X,'POTENTIAL SOLUTION (FT) BIAS= 'F10.3,/)
0349    115     FORMAT(50X,'THICKNESS (FT) (+=UNCONF. -=CONF.)'/)
0350    116     FORMAT(55X,'Q CHECK (CFD)'/)
0351    117     FORMAT(55X,'TRANSMISSIVITY'/)
0352    1117    FORMAT(1H+,71X,'FT**2/DAY')
0353    1127    FORMAT(1H+,71X,'GAL/DAY-FT')
0354    118     FORMAT(55X,'UNBIASED POTENTIAL (FT)'/)
0355    119     FORMAT(55X,'INTERAQUIFER TRANS. (CFD)'/)
0356    121     FORMAT(1X,'TOTAL FLOW (GPM) =',F10.2)
0357    122     FORMAT(55X,'AV. FLUX MAG. (CFD)'/)
0358    123     FORMAT(55X,'FLUX RATIO  (PERCENT)'/)
0359    124     FORMAT(1X,I5,2X,12F10.4)
0360    125     FORMAT(55X,'OCEAN TRANSFER (CFD)'/)
0361            END
```

```
                                                  .TITLE MXSING
 000000                                           .PSECT MATRX,RW,D,OVR,GBL
                                                  .GLOBL FWARD,BKWARD,SHIFT1,SHIFT2
 000000                                   A:      .BLKW   20480.
 120000   000000                         NSTRT:  .WORD
 000000                                           .PSECT  BKA,RW,D,OVR,GBL
 000000   000000                          NE:     .WORD
 000002   000000                          NEQ:    .WORD
 000004   000000                          M:      .WORD
 000006   000000                          N:      .WORD
 000010   000000                          NBF:    .WORD
 000000                                           .PSECT PRGM,RW,I,CON,LCL
 000000   000000                          NSREL:  .WORD
 000002   000000                          NSOLV:  .WORD
 000004   000000                          I0:     .WORD
 000006   000000                          IR:     .WORD
 000010   000000                          J0:     .WORD
 000012   000000                          NLM:    .WORD
 000014   000000                          I0J0:   .WORD
 000016   000000                          BSV:    .WORD
                                          ;
          000000                          F0=R0
          000001                          F1=R1
          000002                          F2=R2
          000003                          F3=R3
          000004                          F4=R4
                                          ;
```

```
000020  017567  000002  177752  FWARD:  MOV     @2(R5),NSREL
000026  017567  000004  177746          MOV     @4(R5),NSOLV
000034  170002                          SETI
000036  170001                          SETF
000040  016700  000004*                 MOV     M,R0
000044  005200                          INC     R0
000046  010067  177732                  MOV     R0,I0
000052  006300                          ASL     R0
000054  010067  177726                  MOV     R0,IB
000060  016767  177714  177722          MOV     NSREL,J0
000066  016700  177710                  MOV     NSOLV,R0                ;CTR NSOLV
000072  016701  177712          LP20:   MOV     J0,R1
000076  005301                          DEC     R1
000100  072127  000011                  ASH     #9.,R1
000104  062701  000000*                 ADD     #A,R1                   ; J0 ADD
000110  016702  177670                  MOV     I0,R2
000114  006302                          ASL     R2
000116  006302                          ASL     R2
000120  060102                          ADD     R1,R2
000122  010267  177666                  MOV     R2,I0J0
000126  172762  177774                  LDF     -4(R2),F3               ; A(I0,J0) TO F3
000132  016702  177650                  MOV     IB,R2
000136  006302                          ASL     R2
000140  006302                          ASL     R2
000142  060102                          ADD     R1,R2
000144  172462  177774                  LDF     -4(R2),F0               ; A(IB,J0) TO F0
000150  [illegible]                     DIVF    F3,F0
000152  [illegible]  177774             STF     F0,-4(R2)
000156  016702  177626                  MOV     J0,R2
[illegible]  [illegible]  [illegible]   ADD     NSTRT,R2
```

```
000166  005302                          DEC     R2
000170  020267  000006'                 CMP     R2,N
000174  002401                          BLT     1$
000176  000207                          RTS     PC
000200  005402                  1$:     NEG     R2
000202  066702  000006'                 ADD     N,R2
000206  010267  177600                  MOV     R2,NLM
000212  020267  000004'                 CMP     R2,M
000216  002403                          BLT     2$
000220  016767  000004' 177564          MOV     M,NLM
000226  016702  177560          2$:     MOV     NLM,R2          ; SET CTR DO 5
000232  016703  177546                  MOV     I0,R3
000236  006303                          ASL     R3
000240  006303                          ASL     R3
000242  060103                          ADD     R1,R3           ; R3 =(M+2,J0)
000244  172413                  LP5:    LDF     (R3),F0
000246  174403                          DIVF    F3,F0
000250  174023                          STF     F0,(R3)+
000252  077204                          SOB     R2,LP5
000254  016702  177526                  MOV     I8,R2
000260  005302                          DEC     R2
000262  006302                          ASL     R2
000264  006302                          ASL     R2
000266  060102                          ADD     R1,R2
000270  172612                          LDF     (R2),F2
000272  174204                          STF     F2,F4
000274  010267  177516                  MOV     R2,BSV
```

```
000300  016702  177506          MOV     NLM,R2          ; OUTER CTR DO 15
000304  016703  177504          MOV     I0J0,R3
000310  062703  177774          ADD     #-4,R3
000314  062703  000774   LP15:  ADD     #508.,R3
000320  172713                  LDF     (R3),F3         ; A(KJ1,KJ)
000322  016701  177466          MOV     I0J0,R1
000326  010304                  MOV     R3,R4
000330  062704  000004          ADD     #4,R4
000334  016705  177452          MOV     NLM,R5          ; INNER CTR DO 10
000340  172421           LP10:  LDF     (R1)+,F0        ; A(KI1,J0)
000342  172514                  LDF     (R4),F1
000344  171003                  MULF    F3,F0
000346  173100                  SUBF    F0,F1
000350  174124                  STF     F1,(R4)+
000352  077506                  SOB     R5,LP10
000354  016704  177432          MOV     NLM,R4
000360  160204                  SUB     R2,R4
000362  005204                  INC     R4
000364  072427  000011          ASH     #9.,R4
000370  066704  177422          ADD     BSV,R4          ; (NLM-R2+1)*512 +R1
000374  172514                  LDF     (R4),F1
000376  172604                  LDF     F4,F2
000400  171203                  MULF    F3,F2
000402  173122                  SUBF    F2,F1
000404  174114                  STF     F1,(R4)
000406  077236                  SOB     R2,LP15
000410  005267  177374          INC     J0
000414  [illegible]             DEC     R0
000416  [illegible]             BGT     LP20
000420  [illegible]             RTS     PC
```

```
                                        ;
                                        ;
000422  017567  000002  177350  BKWARD:  MOV     #2(R5),NSREL
000430  017567  000004  177344          MOV     #4(R5),NSOLV
000436  170002                          SETI
000440  170001                          SETF
000442  016700  177332                  MOV     NSREL,R0
000446  005300                          DEC     R0
000450  010067  177334                  MOV     R0,J0
000454  005300                          DEC     R0
000456  072027  000011                  ASH     #9.,R0
000462  062700  000000'                 ADD     #A,R0                   ; J0
000466  016701  177310                  MOV     NSOLV,R1
000472  005301                          DEC     R1
000474  016702  120000'         LPB10:  MOV     NSTRT,R2
000500  166702  000010'                 SUB     NBF,R2
000504  066702  177300                  ADD     J0,R2
000510  010267  177276                  MOV     R2,NLM
000514  020267  000004'                 CMP     R2,M
000520  002403                          BLT     1$
000522  016767  000004' 177262          MOV     M,NLM
000530  016702  177256          1$:     MOV     NLM,R2                  ; CTR 2
000534  016703  177246                  MOV     I8,R3
000540  005303                          DEC     R3
000542  006303                          ASL     R3
000544  006303                          ASL     R3
000546  060003                          ADD     R0,R3                   ; R3 =(I8,J0)
```

```
000550  172763  001000                LDF     512.(R3),F3
000554  016704  000004'               MOV     M,R4
000560  006304                        ASL     R4
000562  006304                        ASL     R4
000564  005404                        NEG     R4
000566  060304                        ADD     R3,R4                   ; R4 =(I0,J0)
000570  172414                LPB20:  LDF     (R4),F0
000572  172513                        LDF     (R3),F1
000574  171003                        MULF    F3,F0
000576  173100                        SUBF    F0,F1
000600  174113                        STF     F1,(R3)
000602  162704  000774                SUB     #508.,R4
000606  162703  001000                SUB     #512.,R3
000612  077212                        SOB     R2,LPB20
000614  005367  177170                DEC     J0
000620  162700  001000                SUB     #512.,R0
000624  077155                        SOB     R1,LPB10
000626  000207                        RTS     PC
                              ;
                              ;
000630  017500  000002        SHIFT1: MOV     #2(R5),R0
000634  072427  000011                ASH     #9.,R0
000640  012701  000000'               MOV     #A,R1
000644  010102                        MOV     R1,R2
000646  060002                        ADD     R0,R2
000650  012703  000200                MOV     #128.,R3
000654  162703  177126                SUB     IB,R3
000660  006303                        ASL     R3
000662  006303                        ASL     R3
000664  016704  000004'               MOV     M,R4
```

```
000670  016705  177112          SLP:    MOV     I8,R5
000674  012221                  SLP1:   MOV     (R2)+,(R1)+
000676  012221                          MOV     (R2)+,(R1)+
000700  077503                          SOB     R5,SLP1
000702  060301                          ADD     R3,R1
000704  060302                          ADD     R3,R2
000706  077410                          SOB     R4,SLP
000710  000207                          RTS     PC
                                ;
                                ;
000712  017500  000002          SHIFT2: MOV     #2(R5),R0
000716  005300                          DEC     R0
000720  072027  000011                  ASH     #9.,R0
000724  016701  000004'                 MOV     M,R1
000730  010102                          MOV     R1,R2
000732  006302                          ASL     R2
000734  006302                          ASL     R2
000736  062702  000000'                 ADD     #A,R2
000742  072127  000011                  ASH     #9.,R1
000746  060201                          ADD     R2,R1
000750  010102                          MOV     R1,R2
000752  060002                          ADD     R0,R2
000754  016700  000004'                 MOV     M,R0
000760  062700  000002                  ADD     #2,R0
000764  010003                          MOV     R0,R3
000766  062700  000200                  ADD     #128.,R0
000772  006300                          ASL     R0
000774  006300                          ASL     R0
000776  010304                          MOV     R3,R4
001000  005304                          DEC     R4
001002  010305                  US1:    MOV     R3,R5
001004  012122                  US2:    MOV     (R1)+,(R2)+
001006  012122                          MOV     (R1)+,(R2)+
001010  077503                          SOB     R5,US2
001012  160001                          SUB     R0,R1
001014  160002                          SUB     R0,R2
001016  077407                          SOB     R4,US1
001020  000207                          RTS     PC
        000001                          .END
```

```
MXSING  MACRO D1013  30-MAY-79 16:37  PAGE 1-4
SYMBOL TABLE

A       000000R     002 F3     =X000003          LPB20   000570R     004 NBF     000010R     003 SHIFT1  000630RG    004
BKWARD  000422RG    004 F4     =X000004          LP10    000340R     004 NE      000000R     003 SHIFT2  000712RG    004
BSV     000315R     004 IB       000006R     004 LP15    000314R     004 NEQ     000002R     003 SLP     000670R     004
FWARD   000020RG    004 I0       000004R     004 LP20    000072R     004 NLM     000012R     004 SLP1    000674R     004
F0    =X000000          I0J0     000014R     004 LP5     000244R     004 NSOLV   000002R     004 US1     001002R     004
F1    =X000001          J0       000010R     004 M       000004R     003 NSREL   000000R     004 US2     001004R     004
F2    =X000002          LPB10    000474R     004 N       000006R     003 NSTRT   120000R     002

. ABS.  000000     000
        000000     001
MATRX   120002     002
BKA     000012     003
PRGM    001022     004
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  239 WORDS  ( 1 PAGES)
DYNAMIC MEMORY:  11935 WORDS  ( 45 PAGES)
ELAPSED TIME:  00:00:25
,LP:=DB0:MXSING
```

```
DB0:ITTER=ITTER
[1,1]8NWLI8/LB
/
ASG=DB0:2
ASG=LP:6
ASG=TI:5
/
```

```
        C ******* ITTER ******
        C ******* PROGRAM TO RETRIVE PROGRAM RUNNING STATISTICS
        C ******* FROM THE VTT FILEQ DISK
        C ITTER=ITTER,[1,1]BNWLIB,DLB/LB
        C /
        C ASG=DB0:2
        C ASG=LP:6
        C ASG=TI:5
        C /
0001            BYTE STAT,TITLE,TIT,TEST,DEVIN(4),DEVOUT(4)
0002            INTEGER RTIMES,RTIM2,TIMVAL,QPTR,QTIME,RTIME0,RTIME
0003            COMMON/BK0/     I1,I2,NLINES,NCOLS,N1(4),BIAS,I7,I8,IROT,I9,
                1               DELX,N2(30),TITLE(80),N3(9),XLLH,YLLH,FIL(100)
0004            COMMON /RESULT/NUMSAV,RTIMES(3,100),RMIN(100),NTPLSV,ISAVNO(100)
0005            COMMON /STATS/ITPNM,NSTPS,OMEG,CONVG,HIINT,IYHI,IXHI,
                1IAHT,NITER,NSTP,IGX,TTMS,DTMS,HIDF(125)
0006            WRITE(5,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0007            READ(5,*) IUN
0008            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0009            ISUM=0
0010            ISTPSM=0
0011            ITPL=0
0012            DIFSUM=0.
0013            STATS=0.
0014            WRITE(5,555)
0015    555     FORMAT(' LARGE REGION OR SMALL (0 OR 1)')
0016            READ(5,556) IREG
0017    556     FORMAT(I5)
0018            IOF1=0
0019            IOF2=0
0020            IF(IREG.NE.1) GO TO 557
0021            IOF1=100
0022            IOF2=20003.
0023    557     CONTINUE
0024            IF(IREG.NE.1) CALL OPFILE(2,'VTTFIELDS',2401.,IPTR)
0025            IF(IREG.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,IPTR)
0026            CALL OPR(IPTR,0.,I1,100.)
0027            WRITE(6,1)TITLE
0028    1       FORMAT(1H1/15X,80A1/42X,5X,
                1'OMEGA',' CONVERGENCE','    HI DIFF.','    X    Y   AQ',
                1'  NU. ITTER.  TOTAL TIME  DELTA TIME'//)
0029            CALL OPFILE(2,'VTTSTATS',200.,ISTATS)
0030            CALL OPFILE(2,'VTTRESULT',40006.,IANS)
0031            CALL OPR(IANS,0.,+IOF2,NUMSAV,768.)
0032            TTIME=0.
0033    2000    STATS=STATS+1.
0034            CALL OPR(ISTATS,IOF1+STATS,ITPNM,256.)
0035            IF(ITPNM.LT.ITPL.OR.ITPNM.GT.NUMSAV) GO TO 1000
0036            ITPL=ITPNM
0037            ISUM=ISUM+NITER
0038            ISTPSM=ISTPSM+1
0039            DIFSUM=HIINT+DIFSUM
0040            T=TTMS/60.
0041            TL=TTMS
0042            TTIME=TTIME+DTMS
0043            TTIME=TTIME/60.
```

```
0044            WRITE(6,2) NSTP,NSTPS,ITPNM,OMEG,CONVG,MIINT,IXHI
                1,IYHI,IAHI,NITER,TTIMEM,DTMS
0045    2       FORMAT(' STEP',I4,' OF',I4,' STEPS FOR TIME PLANE',I4
                1,F10.3,F12.5,F12.5,3I5,I12,F12.4,F12.2)
0046            GO TO 2000
0047    1000    CONTINUE
0048            SEC=TTIME/ISUM
0049            WRITE(6,3) ISUM,NUMSAV,NTPLSV,ISTPSM,DIFSUM,SEC
0050    3       FORMAT(' TOTAL ITTERATIONS =',I12/
                1' TOTAL NUMBER OF TIME STEPS (INCLUDING T0) SAVED =',I12/
                1' TOTAL NUMBER OF TIME PLANES SAVED (INCLUDING T0) =',I12/
                1' TOTAL NUMBER OF STEPS =',I12/
                1' SUM OF THE HI DIFFERENCES =',F15.5/
                1' AVERAGE NUMBER OF SECONDS PER ITTERATIONS=',F12.4//1H1)
0051            STOP
0052            END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

READN

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DB0:READN,LP:/SH=READN/MP
ASG=DR0:2,DP0:1:3:4:5
UNITS=7
ACTFIL=5
ASG=LP0:7
ASG=TI:6
/
```

```
A:      .FCTR   READN
B:      .FCTR   [1,1]BNWLIB,DLB/LB
C:      .FCTR   PUTEM
D:      .FCTR   GETEM
E:      .FCTR   SVTPLN
        .ROOT   A-B-*(C,D,E)
        .END
```

```
        C*********** READN.FTN  ***********
        C*********** PROGRAM TO RETRIVE THE TIME PLANES FROM THE
        C*********** VTT FILEQ DISK AND STORE INTO BINARY FILES
        C READN,LP:/SH=READN.ODL/MP
        C ASG=DB0:2,DP0:1:3:4
        C ASG=TI: 6
        C /
        C
        C
        C        THIS PROGRAM READS THE VTT MODEL SOLUTION
        C
0001            BYTE DOIT,DOIT1,TITLE
0002            INTEGER DXFER,TRNOSS,YEAR,DAY,HOUR,YEARL,DAYL,HOURL,AUTO,RTIMES
0003            DIMENSION IBUF(100),ARR(5),ARG(4),NDTYP(2,128)
0004            DIMENSION NODLOC(256),NODSTY(256),NTP(2,128),NTP1(2,128)
                1,IOUT(256),FILIN(7),BUF(128),BUFF(128)
0005            DIMENSION TRNSL(128),TOPSL(128)
0006            COMMON /INP/A(4608),OUT(128),NAQS,NODESX,NODESY
                1,IROT,BIAS,IPTR,IPTR2,IPTR3,IPTR4,FILE(7,4),DELX2,IRESOF
0007            COMMON /BK0/ NUNK,IHBW,NLINES,NCOLS,NITT,ITTCNT,CONV,
                1BIASS,NQS,DXFER,IRT,MAXIT,DELX,AUTO,OMEGA,NTMPLN,
                2NOTIMS,TRNOSS,IWID,TIME,STIME,ETIME,YEAR,MONTH,DAY,HOUR,
                3MINN,YEARL,MONTHL,DAYL,HOURL,MINL,NDSX,NDSY,NAQFRS,
                4NUCONF,DOIT,DOIT1,STOMUL,TITLE(80)
                5,COFLEK,SEALEV,DELLEK,ITVN,ILGSML,ISMLSM,
                6XLLH,YLLH,IFAKE
0008            COMMON /RESULT/NUMSAV,RTIMES(3,100),RMIN(100),NTPLSV,ISAVNO(100)
0009            EQUIVALENCE (NUNK,IBUF)
0010            EQUIVALENCE (A(1),NDTYP(1,1)),(A(129),NODLOC(1)),(A(257),
                1NODSTY(1)),(A(355),NTP(1,1)),(A(513),NTP1(1,1)),(A(641),IOUT(
                1128)),(A(769),BUF(1)),(A(897),BUFF(1)),(A(1025),TRNSL(128)),
                1(A(1153),TOPSL(1))
0011            DATA ARG/1.,0.5,.375,.625/
0012            DATA ARR/1.,2.,7.,3.,4./
        C
        C
        C------READ HEADER INFORMATION--------
        C------AND SET DISK FILES------------
        C
        C
0013            I=IFLD(0,3,ITYPE)
0014            CALL SFLD(0,3,I,ITYPE)
0015            WRITE(6,*) 'PROGRAM READN'
0016            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0017            READ(6,*) IUN
0018            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0019    637     CONTINUE
0020            CALL OPFILE(2,'VTTFIELDS',2401.,IPNT)
0021            WRITE(6,769)
0022    769     FORMAT(' LARGE FIELD OR SMALL (0 OR 1)')
0023            READ(6,768) IREGON
0024    768     FORMAT(I5)
0025            IF(IREGON.NE.1) WRITE(6,7768)
0026    7768    FORMAT(20X,'LARGE FIELD')
0027            ISOLOP=0.
0028            IRESOP=0.
```

```
0029              IF(IREGON.NE.1) GO TO 767
0030              WRITE(6,7769)
0031      7769    FORMAT(20X,'SMALL FIELD')
0032              IRESOF=20003.
0033              CALL OPFILE(2,'VTTSMALL',2401.,IPNT)
0034      767     CONTINUE
0035              CALL OPR(IPNT,0.,IBUF,100.)
0036              NAQS=NQS
0037              BIAS=BIASS
0038              DELX2=DELX*DELX
0039              IROT=IRT
0040              NODESX=NDSX
0041              NODESY=NDSY
0042              DELX22=DELX*DELX
0043              CALL OPFILE(2,'BOTTOMTRN',400.,IPTR2)
0044              CALL OPFILE(2,'VTTTEMP0',400.,IPTR3)
0045              CALL OPFILE(2,'VTTTEMP1',400.,IPTR4)
0046              IPTR=IPNT
          C
          C
          C---------FIND OUT WHAT IS REQUESTED--------
          C
          C
0047              WRITE(6,770) (TITLE(I),I=1,71)
0048      770     FORMAT(1X,71A1/' (2)   SAVE CALCULATED TIME PLANES'
                  1' AND DRAWDOWN FILES'/
                  1' (1)   RETRIVE CURRENT SURFACE AND SAVE IT '/
                  1' (0)   RESTORE A STARTING SURFACE '/
                  1' (-1)  EDIT THE HEADER BLOCK  ')
0049              READ(6,771) ICTL
0050      771     FORMAT(I5)
0051              IF(ICTL.EQ.0) GO TO 1000
0052              IF(ICTL.EQ.2) GO TO 3000
0053              IF(ICTL.EQ.-1) GO TO 4000
          C
          C
          C-----READ SOLUTION FROM VTTFIELDS  FILE Q--------
          C-----AND SAVE IT ON RSX11-D FILES----------------
          C
          C
0054              WRITE(6,20030)
0055      20030   FORMAT(' WHICH SURFACE DO YOU WISH TO RETRIVE?'/
                  1' POT=1,BOT=2,TOP=3,TRN=4,CAL=5,STO=6,TRCF=7,OTRCF=8,QS',
                  1'=9')
0056              READ(6,10032) IKND
0057              WRITE(6,20033)
0058              DO 797 I=1,NAQS
0059              IF(IKND.EQ.5.AND.I.NE.1) GO TO 797
0060              READ(6,20034) (FILE(J,I),J=1,7)
0061              N=ICHR(FILE(1,I))
0062      797     CONTINUE
0063              IF(IKND.LT.1.OR.IKND.GT.9) STOP
0064              GO TO (701,702,703,704,705,706,707,708,709),IKND
          C POTENTIAL
0065      701     CALL PUTEM(1.,BIAS,1.,1)
0066              GO TO 710
```

```
        C BOTTOM
0067    702     CALL PUTEM(2.,BIAS,1.,2)
0068            GO TO 710
        C TOP
0069    703     CALL PUTEM(7.,BIAS,1.,3)
0070            GO TO 710
        C TRANSMISSIVITY OR HYDRAULIC CONDUCTIVITY
0071    704     CALL PUTEM(3.,0.,(1./7.481),4)
0072            GO TO 710
        C CALCULATIONAL TYPES
0073    705     CALL PUTEM(5.,0.,1.,5)
0074            GO TO 710
        C STORAGE COEFFICIENT
0075    706     CALL PUTEM(11.,0.,1.,6)
0076            GO TO 710
        C TRANSFER COEFFICIENT
0077    707     CALL PUTEM(8.,0.,1.,7)
0078            GO TO 710
        C OCEAN TRANSFER COEFFICIENT
0079    708     CALL PUTEM(9.,0.,1.,8)
0080            GO TO 710
        C G'S
0081    709     CALL PUTEM(4.,0.,1.,9)
0082    710     CONTINUE
0083            GO TO 737
        C
        C
        C------RESTORE THE SOLUTION FROM THE SOLUTION FILE --------
        C
        C
0084    1000    CONTINUE
0085            WRITE(6,20031)
0086    20031   FORMAT(' WHICH SURFACE DO YOU WISH TO RESTORE?'/
                1' POT=1,BOT=2,TOP=3,TRN=4,CAL=5,STO=6,TRCF=7,OTRCF=8,QS',
                1'=9,LEAK=10,OLD POT=11')
0087            READ(6,10032) IKND
0088    10032   FORMAT(I5)
0089            WRITE(6,20033)
0090    20033   FORMAT(' ENTER INPUT FILE NAME FOR EACH AQUIFER EXCEPT'
                1' FOR CAL=5')
0091            DO 798 I=1,NAQS
0092            IF(IKND.EQ.5.AND.I.NE.1) GO TO 798
0093            READ(6,20034) (FILE(J,I),J=1,7)
0094    20034   FORMAT(7A4)
0095            N=ICHR(FILE(1,I))
0096    798     CONTINUE
0097            IF(IKND.LT.1.OR.IKND.GT.11) STOP
0098            GO TO (801,802,803,804,805,806,807,808,809,810,811),IKND
        C INPUT POTENTIALS AND OUTPUT TO FILE ROTATING IF NECESSARY
0099    801     CALL GETEM(1.,BIAS,1.,1)
0100            GO TO 812
        C NOW DO THE BOTTOMS
0101    802     CALL GETEM(2.,BIAS,1.,2)
0102            GO TO 812
        C NOW DO THE TOPS
0103    803     CALL GETEM(7.,BIAS,1.,3)
```

```
0104            GO TO 812
        C NOW DO THE TRANSMISSIVITY
0105    804     CALL GETEM(3.,0.,(1./7.481),4)
0106            GO TO 812
        C NOW DO THE CALCULATIONAL TYPES
0107    805     CALL GETEM(5.,0.,1.,5)
0108            GO TO 812
        C NOW DO THE STORAGE COEFFICIENTS
0109    806     CALL GETEM(11.,0.,1.,6)
0110            GO TO 812
        C NOW DO THE INTERAQUIFER TRANSFER COEFFICIENTS
0111    807     CALL GETEM(8.,0.,1.,7)
0112            GO TO 812
        C NOW DO THE OCEAN TRANSFER COEFFICIENTS
0113    808     CALL GETEM(9.,0.,1.,8)
0114            GO TO 812
        C NOW THE Q'S
0115    809     CALL GETEM(4.,0.,1.,9)
0116            GO TO 812
        C NOW THE LEAKAGE SURFACE
0117    810     CALL GETEM(6.,BIAS,1.,10)
0118            GO TO 812
        C NOW THE OLD POTENTIAL
0119    811     CALL GETEM(10.,BIAS,1.,11)
0120    812     CONTINUE
        C       ALL CALCULATIONAL TYPES AVAILABLE
0121            IF(IROT.EQ.1) GO TO 7010
        C NO ROTATION
0122            NY=NODESY
0123            NX=NODESX*NAQS
0124            NLNS=NY
0125            NCLS=NX
0126            IAQW=NODESX
0127            GO TO 7020
        C ROTATION
0128    7010    NX=NODESY*NAQS
0129            NY=NODESX
0130            NCLS=NX
0131            NLNS=NY
0132            IAQW=NODESY
0133    7020    CONTINUE
0134            IF(IKND.NE.5) GO TO 401
0135            NMAX=0
0136            NDACT=0
0137            DO 60 ILIN=1,NY
0138            NDPER=0
0139            RB=(ILIN-1)*12.+5.
0140            CALL DPR(IPTR,RB,NODSTY,256.)
0141            DO 61 I=1,128
0142            NODLOC(I*2)=0
0143            NODLOC(I*2-1)=0
0144            NDTYP(1,I)=0
0145            NDTYP(2,I)=0
0146    61      CONTINUE
0147            DO 610 ICOL=1,NX
0148            ITYP=NODSTY(ICOL)
```

```
0149            NDTYP(2,ICOL)=ITYP
0150            IF(ITYP.EQ.1.OR.ITYP.EQ.2) GO TO 616
0151            NOACT=NOACT+1
0152            NOPER=NOPER+1
0153            NLOC=NOPER+1
0154            NDTYP(1,ICOL)=NOACT
0155            NODLOC(NLOC)=ICOL
0156    616     CONTINUE
0157            IF(NOPER.GT.NMAX) NMAX=NOPER
0158            NODLOC(1)=NOPER
0159            CALL DPW(IPTR,RB,NDTYP,256.)
0160            CALL DPW(IPTR,RB+1.,NODLOC,256.)
0161    60      CONTINUE
        C CALCULATE THE HALFBAND WIDTH
0162            IMAX=0
0163            DO 400 ILN=1,NY-1
0164            RB=(ILN-1)*12.+5.
0165            RB1=RB+12.
0166            CALL DPR(IPTR,RB,NTP,256.)
0167            CALL DPR(IPTR,RB1,NTP1,256.)
0168            DO 400 IP=1,NX
0169            IN0=NTP(1,IP)
0170            IN1=NTP1(1,IP)
0171            IF(IN0.EQ.0.OR.IN1.EQ.0) GO TO 400
0172            IDF=IN1-IN0
0173            IF(IDF.GT.IMAX) IMAX=IDF
0174    400     CONTINUE
0175            IHBW=IMAX
0176            NUNK=NOACT
0177            CALL DPW(IPNT,0.,IBUF,100.)
        C DO THE AREA AND NODE TYPE ADJUSTMENTS TO THE TRANSFER COEFFICIENTS
0178    401     IF(IKND.EQ.7.OR.IKND.EQ.8) GO TO 402
0179            GO TO 403
0180    402     CONTINUE
0181            DELX2=DELX**2
0182            DO 92 ILIN=1,NY
0183            RB=(ILIN-1)*12.+5.
0184            RB1=RB+3.
0185            RB2=RB+4.
0186            CALL DPR(IPTR,RB,IOUT,256.)
0187            CALL DPR(IPTR,RB1,TRNSL,256.)
0188            CALL DPR(IPTR,RB2,TOPSL,256.)
0189            DO 94 IAW=1,NAQS
0190            DO 94 IWD=1,IAQW
0191            ICOL=IWD+(IAQ-1)*IAQW
0192            IFRM=ICOL+IAQW
0193            IF(IAQ.EQ.3) IFRM=ICOL-2*IAQW
0194            IF(IAQ.EQ.2) IFRM=ICOL+IAQW
0195            NFRM=IOUT(2*IFRM)
0196            NNOD=IOUT(2*ICOL)
0197            ARF=ARG(NNOD/8+1)
0198            ARFF=ARG(NFRM/8+1)
0199            IF(NFRM.EQ.1) ARFF=0.
0200            IF(NNOD.EQ.1) ARF=0.
0201            ARFF=MIN(ARFF,ARF)
0202            TRNSL(ICOL)=TRNSL(ICOL)*ARFF*DELX2
```

```
0203            TOPSL(ICOL)=TOPSL(ICOL)*ARF*DELX2
0204    94      CONTINUE
0205            CALL DPW(IPTR,RB1,TRNSL,256.)
0206            CALL DPW(IPTR,RB2,TOPSL,256.)
0207    92      CONTINUE
0208    403     CONTINUE
0209            GO TO 737
        C SAVE TIME PLANES AND DRAWDOWN FILES
0210    3000    CALL SVTPLN
        C
        C--------- EXIT OR CONTINUE-----------
        C
        C
0211    737     CONTINUE
0212            WRITE(6,734)
0213    734     FORMAT(' EXIT (0) CONTINUE (1)')
0214            READ(6,771) ICTL
0215            IF(ICTL.EQ.0) STOP
0216            GO TO 637
        C
        C
        C-----------EDIT THE HEADER BLOCK-----------
        C
        C
0217    4000    CONTINUE
0218            CALL DPR(IPNT,0.,IBUF,100.)
0219    4200    WRITE(6,4001) CONV,OMEGA,DOIT1,ITVN,ILGSML,ISMLSM
0220    4001    FORMAT(' EDIT THE HEADER BLOCK (0=DONE) (#=VARIABLE)'/
                1' (1) CONV, CONVERGENCE LIMIT =',F12.6/
                2' (2) OMEGA, OVER RELAXATION FACTOR =',F10.3/
                3' (3) DOIT1, LEAKAGE (1) NO LEAKAGE (0) =',I5/
                4' (4) ITVN, TIME VARING NODES (YES=1, 0=NO) =',I5/
                5' (5) ILGSML, FIELD (0=LARGE, 1=SMALL) =',I5/
                6' (6) ISMLSM, STORE B.C. FOR SMALL FIELD (1=Y,0=N)',I5/
                1)
0221            READ(6,4002) ICTL
0222    4002    FORMAT(I5)
0223            IF(ICTL.EQ.0)  CALL DPW(IPNT,0.,IBUF,100.)
0224            IF(ICTL.EQ.0) GO TO 737
0225            IF(ICTL.GT.6) GO TO 4200
0226            GO TO (4101,4102,4103,4104,4105,4106),ICTL
0227    4101    READ(6,4111) CONV
0228    4111    FORMAT(F10.0)
0229            GO TO 4200
0230    4102    READ(6,4111) OMEGA
0231            GO TO 4200
0232    4103    READ(6,4113) DOIT1
0233    4113    FORMAT(I5)
0234            GO TO 4200
0235    4104    READ(6,4113) ITVN
0236            GO TO 4200
0237    4105    READ(6,4113) ILGSML
0238            GO TO 4200
0239    4106    READ(6,4113) ISMLSM
0240            GO TO 4200
0241            END
```

```
0001            SUBROUTINE GETEM(POS,BIAS,SFACT,IKND)
0002            DIMENSION ICOLG(2,128),IA(8000),IOT(256),VAL(200),IVAL(400)
0003            COMMON /GET/TRNSL(128),IOUT(256)
0004            COMMON /INP/A(4608),OUT(128),NAQS,NODESX,NODESY
                1,IROT,BSSS,IPTR,IPTR2,IPTR3,IPTR4,FILE(7,4),DELX22,IRESOF
0005            EQUIVALENCE (TRNSL(1),ICOLG(1,1))
0006            EQUIVALENCE (A(1),IA(1)),(VAL(1),IVAL(1))
0007            EQUIVALENCE (OUT(1),IOT(1))
0008            IODD=0
0009            DO 10 IAQ=1,NAQS
        C READ IN THE UNROTATED ARRAY FOR THE CURRENT AQUIFER
        C NORMAL INPUT FROM LUN 2
0010    1       CONTINUE
0011            IUN=1
0012            IF(IKND.NE.5) GO TO 3
0013            IF(IAQ.EQ.1) OPEN(UNIT=1,NAME=FILE(1,IAQ),TYPE='OLD',FORM=
                1'FORMATTED',READONLY)
0014            GO TO 2
0015    3       CONTINUE
0016            OPEN(UNIT=1,NAME=FILE(1,IAQ),TYPE='OLD',FORM=
                1'UNFORMATTED',READONLY)
0017    30      NLIN=4608/NODESX
0018            IF(NLIN.GT.NODESY)NLIN=NODESY
0019            NPGS=NODESY/NLIN
0020            IF(MOD(NODESY,NLIN).NE.0)NPGS=NPGS+1
0021            DO 4 IPG=1,NPGS
0022            ISTLN=(IPG-1)*NLIN+1
0023            ISTPLN=ISTLN-1+NLIN
0024            IF(ISTPLN.GT.NODESY)ISTPLN=NODESY
0025            DO 44 ILIN=1,ISTPLN-ISTLN+1
0026            ADR=(ISTLN+ILIN-2)*2.
0027            IDX1=(ILIN-1)*NODESX+1
0028            IDX2=ILIN*NODESX
0029            IF(IODD.EQ.0)READ(1)(A(I),I=IDX1,IDX2)
0030            RNWD=NODESX*2.
0031            IF(IODD.EQ.1)CALL DPR(IPTR4,ADR,A(IDX1),RNWD)
0032            IF(IODD.EQ.2)CALL DPR(IPTR4,ADR,IA(IDX1),RNWD)
0033    44      CONTINUE
        C ROTATION REQUIRED?
0034            IF(IROT.NE.1)GO TO 444
        C YES ROTATE
0035            WORDS=(NODESY*2.)*NAQS
0036            DO 45 ILIN=1,NODESX
0037            ADR=(ILIN-1)*2.
0038            CALL DPR(IPTR3,ADR,VAL,WORDS)
0039            DO 46 ICOL=ISTPLN,ISTLN,-1
        C       K=(ICOL-1)*NODESX+ILIN
0040            K=(ICOL-ISTLN)*NODESX+ILIN
        C       J=(IAQ-1)*NODESY+ICOL
0041            J=(IAQ-1)*NODESY+NODESY-ICOL+1
0042            IF(IODD.NE.2.AND.A(K).NE.0) A(K)=(A(K)+BIAS)*SFACT
0043            IF(IODD.NE.2)VAL(J)=A(K)
0044            IF(IODD.EQ.2)IVAL(J)=IA(K)
0045    46      CONTINUE
0046            CALL DPW(IPTR3,ADR,VAL,WORDS)
0047    45      CONTINUE
```

```
0048            GO TO 4
        C NO ROTATION
0049    444     CONTINUE
0050            WORDS=(NODESX*2.)*NAQS
0051            DO 445 ILIN=ISTLN,ISTPLN
0052            L=(ILIN-ISTLN)*NODESX
0053            ADR=(ILIN-1)*2.
0054            CALL DPR(IPTR3,ADR,VAL,WORDS)
0055            DO 446 J=(IAQ-1)*NODESX+1,IAQ*NODESX
0056            K=J-(IAQ-1)*NODESX+L
0057            IF(IDDD.NE.2.AND.A(K).NE.0) A(K)=(A(K)+BIAS)*SFACT
0058            IF(IDDD.NE.2)VAL(J)=A(K)
0059            IF(IDDD.EQ.2)IVAL(J)=IA(K)
0060    446     CONTINUE
0061            CALL DPW(IPTR3,ADR,VAL,WORDS)
0062    445     CONTINUE
0063    4       CONTINUE
0064            GO TO 10
        C COLUMN OF CHANGE INPUT OF CALCULATIONAL TYPES FROM LUN IUN
0065    2       CONTINUE
0066            IF(IUN.EQ.6) GO TO 20
0067            READ(IUN,51)
0068    51      FORMAT(3I3)
0069            READ(IUN,51)
0070            READ(IUN,51)
0071    20      CONTINUE
0072            IDDD=2
0073    21      READ(IUN,51) IB,IT,KKK
0074            READ(IUN,52) ((ICOLG(I,JJJ),I=1,2),JJJ=1,KKK)
0075    52      FORMAT(14(I3,02))
0076            DO 53 ICHNG=1,KKK
0077            IS1=ICOLG(1,ICHNG)
0078            IKIND=ICOLG(2,ICHNG)
0079            KKIND=IKIND
0080            IF(IROT.EQ.1) CALL ROTATE(IKIND,KKIND)
0081            IS2=ICOLG(1,ICHNG+1)-1
0082            IF(ICHNG+1.GT.KKK) IS2=NODESX
0083            DO 54 ICOL=IS1,IS2
0084    54      IVAL(ICOL)=KKIND
0085    53      CONTINUE
0086            RNWD=((NODESX+1)/2)*2.
0087            DO 531 ILIN=IB,IT
0088            ADR=(ILIN-1.)*2.
0089            CALL DPW(IPTR4,ADR,IVAL,RNWD)
0090    531     CONTINUE
0091            IF(IT.LT.NODESY) GO TO 20
0092            GO TO 30
0093    10      IF(IKND.NE.5) CLOSE(UNIT=IUN)
        C COPY BOTTOMIRN TO VTTFIELDS
0094            SIZ=12.
0095            NLS=NODESY
0096            NCL=NODESX*NAQS
0097            IF(IROT.EQ.0)GO TO 11
0098            NLS=NODESX
0099            NCL=NODESY*NAQS
0100    11      CONTINUE
```

```
FORTRAN IV-PLUS V02-51D          16:41:39     30-MAY-79            PAGE 3
GETEM.FTN        /TR:BLOCKS/WR

0101             DO 12 IL=1,NLS
0102             ADR1=(IL-1)*2.
0103             ADR2=(IL-1)*SIZ+POS
0104             CALL DPR(IPTR3,ADR1,VAL,NCL*2.)
0105             CALL DPW(IPTR,ADR2,VAL,NCL*2.)
0106     12      CONTINUE
0107             IF(IKND.EQ.5) CLOSE(UNIT=IUN)
0108             RETURN
0109             END
```

```
FORTRAN IV-PLUS V02-51D          16:42:19     30-MAY-79
GETEM.FTN        /TR:BLOCKS/WR

0001             SUBROUTINE ROTATE(ITYPE,KTYPE)
0002             IF(ITYPE-3)99,99,1
0003     1       IROT=IFLD(0,3,ITYPE)
0004             IF(IROT.EQ.0)GO TO 5
0005             IF(IROT.EQ.1)GO TO 10
0006             IROT=IROT-2
0007             GO TO 20
0008     5       IROT=6
0009             GO TO 20
0010     10      IROT=7
0011     20      CALL SFLD(0,3,IROT,ITYPE)
0012     99      KTYPE=ITYPE
0013             RETURN
0014             END
```

```
0001            SUBROUTINE PUTEM(POS,BIAS,SFACT,IKND)
0002            DIMENSION ICOLG(2,128),IA(8000),IOT(256),VAL(200),IVAL(400),
                1NDTYP(2,128),ARG(4)
0003            COMMON /GET/TRNSL(128),IOUT(256)
0004            COMMON /INP/A(4608),OUT(128),NAQS,NODESX,NODESY
                1,IROT,BSSS,IPTR,IPTR2,IPTR3,IPTR4,FILE(7,4),DELX22,IRESOF
0005            EQUIVALENCE (TRNSL(1),ICOLG(1,1))
0006            EQUIVALENCE (A(1),IA(1)),(VAL(1),IVAL(1))
0007            EQUIVALENCE (OUT(1),IOT(1))
0008            DATA ARG/1.,0.5,.375,.625/
0009            IODD=0
        C
        C
        C---------READ FROM THE FILE Q FILE AND--------
        C---------UNROTATE AQUIFER BY AQUIFER----------
        C
        C
0010            DO 10 IAQ=1,NAQS
0011            IF(IKND.EQ.5) GO TO 100
0012            OPEN(UNIT=1,NAME=FILE(1,IAQ),TYPE='NEW',FORM='UNFORMATTED')
        C----------ROTATED OR NOT?--------
0013            IF(IROT.EQ.1) GO TO 13
        C NO ROTATION REQUIRED READ AND COPY FROM FILE Q FILE
        C ADJUSTING FOR BIAS AND SCALE FACTOR
0014            RNWDS=NODESX*NAQS*2.
0015            DO 11 IY=1,NODESY
0016            ADR=(IY-1)*12.+POS
0017            CALL DPR(IPTR,ADR,VAL,RNWDS)
0018            CALL DPR(IPTR,(IY-1)*12.+5.,NDTYP,RNWDS)
0019            DO 12 IX=1,NODESX
0020            IXX=(IAQ-1)*NODESX+IX
0021            OUT(IX)=0.
0022            IF(NDTYP(2,IXX).NE.1) OUT(IX)=VAL(IXX)/SFACT-BIAS
0023            IF(IKND.EQ.8) GO TO 121
0024            IF(IKND.EQ.7) GO TO 122
0025            GO TO 12
0026    121     IF(NDTYP(2,IXX).EQ.1)GO TO 12
0027            ITYPE=NDTYP(2,IXX)/8+1
0028            OUT(IX)=OUT(IX)/(ARG(ITYPE)*DELX22)
0029            GO TO 12
0030    122     IFRM=IXX+NODESX
0031            IF(IAQ.EQ.3) IFRM=IXX-2*NODESX
0032            IF(IAQ.EQ.2) IFRM=IXX+NODESX
0033            NFRM=NDTYP(2,IFRM)
0034            NNOD=NDTYP(2,IXX)
0035            IF(NFRM.EQ.1.OR.NNOD.EQ.1) GO TO 12
0036            ARF=MIN(ARG(NFRM/8+1),ARG(NNOD/8+1))
0037            OUT(IX)=OUT(IX)/(ARF*DELX22)
0038    12      CONTINUE
0039            WRITE(1) (OUT(I),I=1,NODESX)
0040    11      CONTINUE
0041            CLOSE(UNIT=1)
0042            GO TO 10
        C
        C
        C----------ROTATION REQUIRED----------
```

```
      C
      C
0043  13      CONTINUE
0044          RNWDS=NODESY*NAQS*2.
0045          RNWD=NODESX*2.
0046          NLIN=4608/NODESY
0047          IF(NLIN.GT.NODESX) NLIN=NODESX
0048          NPGS=NODESX/NLIN
0049          IF(MOD(NODESX,NLIN).NE.0) NPGS=NPGS+1
0050          ISPOS=IAQ*NODESY
0051          ISTPOS=(IAQ-1)*NODESY+1
0052          DO 4 IPG=1,NPGS
0053          ISTLN=(IPG-1)*NLIN+1
0054          ISTPLN=ISTLN-1+NLIN
0055          IF(ISTPLN.GT.NODESX) ISTPLN=NODESX
0056          DO 5 ILNR=ISTLN,ISTPLN
0057          ADR=(ILNR-1)*12.+POS
0058          CALL DPR(IPTR,ADR,VAL,RNWDS)
0059          CALL DPR(IPTR,(ILNR-1)*12.+5.,NDTYP,RNWDS)
0060          DO 6 IRX=ISPOS,ISTPOS,-1
0061          IY=(ISPOS-IRX)+1
0062          IX=ILNR-ISTLN+1
0063          IXA=(IY-1)*(ISTPLN-ISTLN+1)+IX

0064          A(IXA)=0.
0065          IF(NDTYP(2,IRX).NE.1)A(IXA)=VAL(IRX)/SFACT-BIAS
0066          IF(IKND.EQ.8) GO TO 120
0067          IF(IKND.EQ.7) GO TO 119
0068          GO TO 6
0069  119     IFRM=IRX+NODESY
0070          IF(IAW.EQ.3) IFRM=IRX-2*NODESY
0071          IF(IAW.EQ.2) IFRM=IRX+NODESY
0072          NFRM=NDTYP(2,IFRM)
0073          NNOD=NDTYP(2,IXX)
0074          IF(NFRM.EQ.1.OR.NNOD.EQ.1) GO TO 6
0075          ARF=MIN(ARG(NFRM/8+1),ARG(NNOD/8+1))
0076          A(IXA)=A(IXA)/(ARF*DELX22)
0077          GO TO 6
0078  120     IF(NDTYP(2,IRX).EQ.1) GO TO 6
0079          ITYPE=NDTYP(2,IRX)/8 +1
0080          A(IXA)=A(IXA)/(ARG(ITYPE)*DELX22)
0081  6       CONTINUE
0082  5       CONTINUE
0083          DO 7 I=1,NODESY
0084          ADR=(I-1)*2.
0085          CALL DPR(IPTR2,ADR,VAL,RNWD)
0086          DO 8 IX=ISTLN,ISTPLN
0087          IXX=IX-ISTLN+1+(I-1)*(ISTPLN-ISTLN+1)
0088  8       VAL(IX)=A(IXX)
0089          IF(IPG.NE.NPGS)CALL DPW(IPTR2,ADR,VAL,RNWD)
0090          IF(IPG.EQ.NPGS)WRITE(1)(VAL(J),J=1,NODESX)
0091  7       CONTINUE
0092  4       CONTINUE
0093          CLOSE (UNIT=1)
0094          GO TO 10
      C
```

```
          C
          C        CALCULATIONAL TYPES
          C
          C
0095      100      CONTINUE
          C      COPY CALCULATIONAL TYPES TO TEMPORARY FILE
0096               IF(IAQ.NE.1)GO TO 101
0097               OPEN(UNIT=1,NAME=FILE(1,IAQ),TYPE='NEW')
0098      101      WRITE(1,102)NODESX,NODESY,NAQS
0099      102      FORMAT(3I5)
0100               WRITE(1,103)
0101      103      FORMAT(' 0.25')
0102               WRITE(1,104)IAQ
0103      104      FORMAT(' AQUIFER #',I5)
          C
          C      WHICH ROTATION
          C
0104               IF(IROT.EQ.1)GO TO 113
          C
          C
          C        NO ROTATION SO JUST COPY
          C
          C
0105               RNWDS=NODESX*NAQS*2.
0106               RNWD=NODESX*2.
0107               DO 105 IY=1,NODESY
0108               ADR=(IY-1)*12.+5.
0109               ADR1=(IY-1)*2.
0110               CALL DPR(IPTR,ADR,NDTYP,RNWDS)
0111               DO 106 I=1,NODESX
0112      106      IVAL(I)=NDTYP(2,I+(IAQ-1)*NODESX)
0113               CALL DPW(IPTR2,ADR1,IVAL,RNWD)
0114      105      CONTINUE
          C        READY TO DECODE
0115               GO TO 150
          C
          C
          C        FOR ROTATED
          C
          C
0116      113      CONTINUE
          D        WRITE(6,1113)
          D1113    FORMAT(' 113 CONTINUE')
0117               NLIN=8000/NODESY
0118               RNWDS=NODESY*NAQS*2.
0119               RNWD=NODESX*2.
0120               IF(NLIN.GT.NODESX)NLIN=NODESX
0121               NPGS=NODESX/NLIN
0122               IF(MOD(NODESX,NLIN).NE.0)NPGS=NPGS+1
0123               ISPOS=IAQ*NODESY
0124               ISTPOS=(IAQ-1)*NODESY+1
0125               DO 114 IPG=1,NPGS
          D        WRITE(6,1115)
          D1115    FORMAT(' DO 114')
0126               ISTLN=(IPG-1)*NLIN+1
0127               ISTPLN=ISTLN-1+NLIN
```

```
0128                IF(ISTPLN.GT.NODESX) ISTPLN=NODESX
0129                DO 115 ILNR=ISTLN,ISTPLN
          D         WRITE(6,1114)
          D1114     FORMAT(' DO 115')
0130                ADR=(ILNR-1)*12.+POS
0131                CALL DPR(IPTR,ADR,NDTYP,RNWDS)
0132                DO 116 IRX=ISPOS,ISTPOS,-1
          D         WRITE(6,1116)
          D1116     FORMAT(' DO 116')
0133                IY=(ISPOS-IRX)+1
0134                IX=ILNR-ISTLN+1
0135                IXA=(IY-1)*(ISTPLN-ISTLN+1)+IX
0136                ITYPE=NDTYP(2,IRX)
0137                CALL UNRUT(ITYPE,KTYPE)
0138                IA(IXA)=KTYPE
0139      116       CONTINUE
0140      115       CONTINUE
0141                DO 117 I=1,NODESY
          D         WRITE(6,1117)
          D1117     FORMAT(' DO 117')
0142                ADR=(I-1)*2.
0143                CALL DPR(IPTR2,ADR,IVAL,RNWD)
0144                DO 118 IX=ISTLN,ISTPLN
          D         WRITE(6,1118)
          D1118     FORMAT(' DO 118')
0145                IXX=IX-ISTLN+1+(I-1)*(ISTPLN-ISTLN+1)
0146      118       IVAL(IX)=IA(IXX)
0147                CALL DPW(IPTR2,ADR,IVAL,RNWD)
0148      117       CONTINUE
0149      114       CONTINUE
          C
          C
          C         DO COLUMN OF CHANGE DECODE
          C
          C
0150      150       CONTINUE
0151                RNWD=NODESX+MOD(NODESX,2)
0152                DO 200 IY=1,NODESY
0153                NCHGS=1
0154                ADR=(IY-1)*2.
0155                CALL DPR(IPTR2,ADR,IOT,RNWD)
0156                NDTYP(1,NCHGS)=1
0157                NDTYP(2,NCHGS)=IOT(1)
0158                DO 210 IX=2,NODESX
0159                IF(IOT(IX).EQ.NDTYP(2,NCHGS)) GO TO 210
0160                NCHGS=NCHGS+1
0161                NDTYP(1,NCHGS)=IX
0162                NDTYP(2,NCHGS)=IOT(IX)
0163      210       CONTINUE
0164                WRITE(1,211)IY,IY,NCHGS
0165      211       FORMAT(3I3)
0166                WRITE(1,212)((NDTYP(1,I),NDTYP(2,I)),I=1,NCHGS)
0167      212       FORMAT(14(I3,O2))
0168      200       CONTINUE
0169      10        CONTINUE
0170                IF(IRND.EQ.5) CLOSE(UNIT=1)
0171                RETURN
0172                END
```

```
0001            SUBROUTINE UNROT(ITYPE,KTYPE)
0002            IF(ITYPE-3)99,99,1
0003     1      IROT=IFLD(0,3,ITYPE)
0004            IF(IROT.EQ.6) GO TO 5
0005            IF(IROT.EQ.7)GO TO 10
0006            IROT=IROT+2
0007            GO TO 20
0008     5      IROT=0
0009            GO TO 20
0010     10     IROT=1
0011     20     CALL SFLD(0,3,IROT,ITYPE)
0012     99     KTYPE=ITYPE
0013            RETURN
0014            END
```

```
          ----------------------------------------
00001          SUBROUTINE SVTPLN
00002    C SUBROUTINE TO SAVE TIME PLANES AND MAKE DRAWDOWN FILES
00003          BYTE DESIG(6),FILOUT(28)
00004          BYTE DOIT,DOIT1,TITLE
00005          INTEGER DXFER,TRNOSS,YEAR,DAY,HOUR,YEARL,DAYL,HOURL,AUTO,RTIMES
00006         1,TIMVAL,QPTR,QTIME
00007    C
00008          COMMON /INP/A(4608),OUT(128),NAQS,NODESX,NODESY
00009         1,IROT,BIAS,IPTR,IPTR2,IPTR3,IPTR4,FILE(7,4),DELX2,IRESOF
00010    C
00011          COMMON /BK0/ NUNK,IHBW,NLINES,NCOLS,NITT,ITTCNT,CONV,
00012         1BIASS,NQS,DXFER,IRT,MAXIT,DELX,AUTO,OMEGA,NTMPLN,
00013         2NOTIMS,TRNOSS,IWID,TIME,STIME,ETIME,YEAR,MONTH,DAY,HOUR,
00014         3MINN,YEARL,MONTHL,DAYL,HOURL,MINL,NDSX,NDSY,NAQFRS,
00015         4NUCONF,DOIT,DOIT1,STOMUL,TITLE(80)
00016         5,COFLEK,SEALEV,DELLEK,ITVN,ILGSML,ISMLSM,
00017         6XLLH,YLLH,IFAKE
00018    C
00019    C SIZE 370. WORDS
00020          COMMON /TIMR/NQTS,NTPLNS,TIMVAL(5,25),RATIO(25),NSTEPS(25)
00021         1,QTIME(5,28),QPTR(28)
00022    C
00023    C SIZE 602. WORDS
00024          COMMON /RESULT/NUMSAV,RTIMES(3,100),RMIN(100),NTPLSV,ISAVNO(100)
00025    C
00026    C DESCRIPTOR DATA 32 WORDS
00027          COMMON /WELL/ WELNAM(7),WELLID(3),WELLX,WELLY,ITIME(5),VAL0
00028         1,IAQNUM
00029    C ACTUAL DATA 108 WORDS
00030          COMMON /WELL1/ VAL(50),ITIMEA(5),ATIME,ITOSS1
00031    C
00032          DIMENSION IWELLX(50),IWELLY(50),IWLAD(50),IWLAQ(50)
00033          DIMENSION WELXY(2,50),IA(9216)
00034          DIMENSION WELLST(7),ILIN(20,200),NWLSPL(200)
00035          EQUIVALENCE (A(1),NWLSPL(1)),(A(101),ILIN(1,1))
00036          EQUIVALENCE (OUT(1),WELXY(1,1)),(A(1),IA(1))
00037    C
00038          QADR=11201.
00039          IF(IRESOF.EQ.20003.) QADR=11203.
00040          CALL OPFILE(2,'VTTQS',11205.,IQPTR)
00041          CALL OPR(IQPTR,QADR,NQTS,370.)
00042    D     WRITE(7,*) 'NSTEPS'
00043    D     WRITE(7,*) NSTEPS
00044          WRITE(6,*) 'DO YOU WANT THE TIME PLANES SAVED 0=NO  1=YES'
00045          READ(6,*) ISAVTP
00046          IF(ISAVTP.EQ.1)
00047         .  WRITE(6,*) 'ENTER A FULL SIX CHARACTER DESIGNATION FOR '
00048         .  WRITE(6,*) 'THE TIME PLANE FILES'
00049         .  READ(6,1) DESIG
00050    1    .  FORMAT(6A1)
00051         ...FIN
00052          WRITE(6,*) 'SAVE THE HYDROGRAPH FILES 0=NO  1=YES'
00053          READ(6,*) IDRNON
```

```
00054          NWELLS=0
00055          IF(IDRWDN.EQ.1)
00056          . CALL DPFILE(2,'VTTTEMP0',400.,IWLL)
00057          . WRITE(6,*) 'ENTER THE NAME OF LIST OF HYDROGRAPHS TO BE'
00058          . WRITE(6,*) 'PREPARED'
00059          . READ(6,2) WELLST
00060    2     . FORMAT(7A4)
00061          . N=ICHR(WELLST)
00062          . OPEN(UNIT=1,NAME=WELLST,TYPE='OLD',READONLY)
00063          . XURH=XLLH+(NDSX-1)*DELX
00064          . YURH=YLLH+(NDSY-1)*DELX
00065          . ITOSS=0
00066          . ITOSS1=0
00067          . VAL0=0.
00068          . ITIME(1)=YEAR
00069          . ITIME(2)=MONTH
00070          . ITIME(3)=DAY
00071          . ITIME(4)=HOUR
00072          . ITIME(5)=MINN
00073          . DO(I=1,200)
00074          . . DO(J=1,20) ILIN(J,I)=0
00075          . . NWLSPL(I)=0
00076          . ...FIN
00077          . READ(1,*) NWLS
00078          . IF(NWLS.GT.50) NWLS=50
00079          . DO(I=1,NWLS)
00080          . . READ(1,2) WELLID
00081          . . READ(1,2) WELNAM
00082          . . N=ICHR(WELNAM)
00083          . . READ(1,3) WELLX,WELLY,IAQNUM
00084    D     . . WRITE(7,300) WELLID,WELNAM,WELLX,WELLY,IAQNUM
00085    D300  . . FORMAT(1X,3A4/1X,7A4/2F12.2,I5)
00086    3     . . FORMAT(2F20.0,I5)
00087          . . IF(IAQNUM.EQ.0) IAQNUM=1
00088          . . IF(WELLX.GE.XLLH.AND.WELLX.LE.XURH)
00089          . . . IF(WELLY.GE.YLLH.AND.WELLY.LE.YURH)
00090          . . . . IF(WELLX.EQ.XURH) WELLX=XURH-1.0E-10
00091          . . . . IF(WELLY.EQ.YURH) WELLY=YURH-1.0E-10
00092          . . . . NWELLS=NWELLS+1
00093          . . . . ADR=NWELLS-1
00094          . . . . CALL DPW(IWLL,ADR,WELNAM,32.)
00095          . . . . ILN=(WELLY-YLLH)/DELX+1
00096          . . . . ICL=(WELLX-XLLH)/DELX+1
00097          . . . . ICLL=ICL
00098          . . . . ILNN=ILN
00099          . . . . WELXY(1,NWELLS)=(WELLX-XLLH)/DELX+1.
00100          . . . . WELXY(2,NWELLS)=(WELLY-YLLH)/DELX+1.
00101          . . . . IF(IRT.EQ.1)
00102          . . . . . ILNN=ICL
00103          . . . . . SAV=WELXY(2,NWELLS)
00104          . . . . . WELXY(2,NWELLS)=WELXY(1,NWELLS)
00105          . . . . . WELXY(1,NWELLS)=NDSY+1.-SAV
00106          . . . . . ICLL=WELXY(1,NWELLS)
00107          . . . . ...FIN
00108          . . . . NWLSPL(ILNN)=NWLSPL(ILNN)+1
00109          . . . . ILTN(NWLSPL(ILNN),ILNN)=ICLL*100+NWELLS
```

```
00110          .  .  .  .  IWLAQ(NWELLS)=IAQNUM
00111    D     .  .  .  .  WRITE(7,*) NWELLS,IRT,ICLL,ILNN,WELXY(1,NWELLS),WELXY(2,NWELLS)
00112    D   1 .  .  .  .  ,NWLSPL(ILNN),ILIN(NWLSPL(ILNN),ILNN),IWLAQ(NWELLS)
00113          .  .  .  .  CONTINUE
00114          .  .  .  ...FIN
00115          .  .  ...FIN
00116          .  ...FIN
00117          .  IAGADR=NOSX
00118          .  IF(IRT.EQ.1) IAGADR=NDSY
00119          .  WHEN(NWELLS.EQ.0) IDRWDN=0
00120          .  ELSE
00121          .  .  K=0
00122          .  .  DO(I=1,200)
00123          .  .  .  IF(NWLSPL(I).NE.0)
00124          .  .  .  .  DO(J=1,NWLSPL(I))
00125          .  .  .  .  .  K=K+1
00126          .  .  .  .  .  IWELLX(K)=ILIN(J,I)/100
00127          .  .  .  .  .  IWELLY(K)=I
00128          .  .  .  .  .  IWLAD(K)=ILIN(J,I)-100*IWELLX(K)
00129    D     .  .  .  .  .  WRITE(7,*) I,J,K,IWELLX(K),IWELLY(K),IWLAD(K)
00130          .  .  .  .  .  CONTINUE
00131          .  .  .  .  ...FIN
00132          .  .  .  ...FIN
00133          .  .  ...FIN
00134          .  ...FIN
00135    D     .  WRITE(7,*) IWELLX
00136    D     .  WRITE(7,*) IWELLY
00137    D     .  WRITE(7,*) IWLAD
00138    D     .  WRITE(7,*) IWLAQ
00139          .  CLOSE(UNIT=1)
00140          ...FIN
00141    C BEGIN RETRIVING THE TIME PLANES
00142          CALL OPFILE(2,'VTTRESULT',40006.,IANS)
00143          CALL OPFILE(2,'UNROTATE',800.,ITMPTR)
00144          CALL OPR(IANS,0.+IRESOF,NUMSAV,602.)
00145    C CODING REQUIRED FOR COMPRESSION OF TIME STEP AND TIME PLANE
00146    C POTENTIAL DATA IN THE RESULT FILE AND THE CODING TO RETRIVE
00147    C THIS COMPRESSED DATA AND PUT INTO THE VTTFIELDS FILE
00148    C
00149    C THIS FOLLOWING PROCEEDURE NEEDS ONLY BE DONE ONCE FOR AND GIVEN
00150    C SYSTEM IT CALCULATES SPACE REQUIRED AND THE NUMBER OF PAGES OF
00151    C DATA NEEDED FOR THE COMPRESSION
00152          NLIN=(4608-128)/NCOLS
00153          IF(NLIN.GT.NLINES) NLIN=NLINES
00154          NPGS=NLINES/NLIN
00155          IF(MOD(NLINES,NLIN).NE.0) NPGS=NPGS+1
00156    C NPG=TOTAL NUMBER OF PAGES REQUIRED
00157    C NLIN=NUMBER OF MATRIX LINES PER PAGE NOT COUNTING THE 1 LINE OVERLAP
00158    C NCOLS=TOTAL NUMBER OF COLUMNS PER MATRIX LINE
00159    C NLINES=TOTAL NUMBER OF MATRIX LINES
00160    C TOTAL SIZE OF BUFFER REQUIRED FOR THIS PROCEEDURE IS 4608 REAL WORDS
00161    C TO ALLOW FOR THE ONE LINE OVERLAP
00162    C NOW CALCULATE THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR STORAGE
00163    C OF A TIME PLANE
00164    C
00165    C ADJUST THE LINE COUNT FOR OVERLAP
```

```
00166          IF(NLIN.NE.NLINES) NLINA=NLIN+1
00167    C CALCULATE THE DISK BLOCKS REQUIRED FOR ONE FULL PAGE
00168          IBLKPP=NLINA*NCOLS/128
00169          IF(MOD(NLINA*NCOLS,128).NE.0) IBLKPP=IBLKPP+1
00170          ITOTBK=IBLKPP*(NPGS-1)
00171    C ITOTBK IS THE BLOCKS REQUIRED FOR FULL PAGES NOW CALCULATE
00172    C THE BLOCKS REQUIRED FOR THE LAST PAGE
00173          LINLFT=NLINES-NLIN*(NPGS-1)
00174          IBLKLF=LINLFT*NCOLS/128
00175          IF(MOD(LINLFT*NCOLS,128).NE.0) IBLKLF=IBLKLF+1
00176          ITOTBK=ITOTBK+IBLKLF
00177    C NOW ITOTBK IS THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR TIME PLANE
00178    C SAVING.
00179          J=0
00180          DO(KSAV=1,NTPLSV)
00181          .  IF(KSAV.EQ.1) NSKP=0
00182          .  IF(J.NE.0) NSKP=NSTEPS(KSAV-1)-1
00183          .  IF(ISAVTP.EQ.1)
00184          .  .  DO(IAQ=1,NAQS)
00185          .  .  .  KSAVV=KSAV-1
00186          .  .  .  IUNT=IAQ+2
00187          .  .  .  WHEN(KSAVV.LT.10)
00188          .  .  .  .  ENCODE(28,298,FILOUT) DESIG,IAQ,KSAVV
00189    298   .  .  .  .  FORMAT(6A1,'PT',I1,'.T0',I1,15X)
00190          .  .  .  ...FIN
00191          .  .  .  ELSE
00192          .  .  .  .  ENCODE(28,299,FILOUT) DESIG,IAQ,KSAVV
00193    299   .  .  .  .  FORMAT(6A1,'PT',I1,'.T',I2,15X)
00194          .  .  .  ...FIN
00195          .  .  .  N=ICHR(FILOUT)
00196          .  .  .  OPEN(UNIT=IUNT,NAME=FILOUT,TYPE='NEW',FORM='UNFORMATTED')
00197          .  .  ...FIN
00198          .  ...FIN
00199    C   READ TIME PLANE DATA
00200          .  IF(IDRWON.EQ.1)
00201          .  .  IWL=1
00202          .  .  DO(I=1,50) VAL(I)=0.
00203          .  .  ITIMEA(1)=RTIMES(1,KSAV)
00204          .  .  ITIMEA(2)=RTIMES(2,KSAV)/100
00205          .  .  ITIMEA(3)=RTIMES(2,KSAV)-ITIMEA(2)*100
00206          .  .  ITIMEA(4)=RTIMES(3,KSAV)/100
00207          .  .  ITIMEA(5)=RTIMES(3,KSAV)-100*ITIMEA(4)
00208          .  .  ATIME=RMIN(KSAV)
00209          .  .  ITUSS1=0
00210          .  ...FIN
00211    C     .
00212          .  J=J+1
00213          .  J=J+NSKP
00214          .  DO(IPG=1,NPGS)
00215          .  .  ISTLN=(IPG-1)*NLIN+1
00216          .  .  ISTPLN=ISTLN+NLIN-1
00217          .  .  ISTPLN=MIN(ISTPLN,NLINES)
00218          .  .  IYMAX=ISTPLN
00219          .  .  IYMIN=ISTLN
00220          .  .  IF(IPG.NE.NPGS) ISTPLN=ISTPLN+1
00221          .  .  ADR=IRESOF+3.+ITOTBK*(J-1)+(IPG-1)*IBLKPP
```

```
00222          .  .  BUFWDS=(ISTPLN-ISTLN+1)*NCOLS*2
00223          .  .  CALL DPR(IANS,ADR,A(1),BUFWDS)
00224          .  .  IF(BIAS.NE.0.)
00225          .  .  .  DO(IP=1,NCOLS*(ISTPLN-ISTLN+1))
00226          .  .  .  .  IF(A(IP).NE.0) A(IP)=A(IP)-BIAS
00227          .  .  .  ...FIN
00228          .  .  ...FIN
00229    C SAVE HYDROGRAPH DATA HERE
00230          .  .  IF(IDRWON.EQ.1)
00231    D     .  .  .  WRITE(7,*) 'SAVE HYDROGRAPH DATA',IPG
00232          .  .  .  DO(ILN=IYMIN,IYMAX)
00233          .  .  .  .  IPOS=(ILN-IYMIN)*NCOLS+1
00234          .  .  .  .  WHILE(IWELLY(IWL).EQ.ILN)
00235          .  .  .  .  .  N=IWLAD(IWL)
00236          .  .  .  .  .  IAQNN=IWLAQ(N)
00237          .  .  .  .  .  IPOSS=IPOS+(IAQNN-1)*IAQADR
00238          .  .  .  .  .  IPOS1=IPOSS+NCOLS
00239          .  .  .  .  .  X=WELXY(1,N)
00240          .  .  .  .  .  Y=WELXY(2,N)
00241    D     .  .  .  .  .  WRITE(7,*) ILN,IWL,IWELLY(IWL),X,Y,N,IAQNN,IPOS,IPOSS,IPOS1
00242          .  .  .  .  .  CALL FGEN2(A(IPOSS),A(IPOS1),IAQADR,IAQADR,NLINES,X,Y,
00243         1.  .  .  .  .  PRED,IFLG)
00244          .  .  .  .  .  IF(IFLG.NE.0) VAL(IWL)=PRED
00245          .  .  .  .  .  IWL=IWL+1
00246    D     .  .  .  .  .  WRITE(7,*) IWL,IFLG,PRED,VAL(IWL)
00247          .  .  .  .  .  CONTINUE
00248          .  .  .  .  ...FIN
00249          .  .  .  ...FIN
00250          .  .  ...FIN
00251          .  .  IF(ISAVTP.EQ.1)
00252          .  .  .  WHEN(IROT.EQ.0)
00253          .  .  .  .  ISTOP=ISTPLN
00254          .  .  .  .  IF(IPG.NE.NPGS) ISTOP=ISTOP-1
00255          .  .  .  .  DO(IL=ISTLN,ISTOP)
00256          .  .  .  .  .  IPOS=(IL-ISTLN)*NCOLS+1
00257          .  .  .  .  .  WRITE(3) (A(K),K=IPOS,IPOS+NDSX-1)
00258          .  .  .  .  .  IF(NAQS.GE.2) WRITE(4) (A(K),K=IPOS+NDSX,IPOS+2*NDSX-1)
00259          .  .  .  .  .  IF(NAQS.GE.3) WRITE(5) (A(K),K=IPOS+NDSX*2,IPOS+NDSX*3-1)
00260          .  .  .  .  ...FIN
00261          .  .  .  ...FIN
00262          .  .  .  ELSE CALL UNROT(A,NDSX,NDSY,NAQS,NPGS,ISTLN,ISTPLN,ITMPTR)
00263          .  .  ...FIN
00264          .  ...FIN
00265    C WRITE OUT THIS SET OF HYDROGRAPH DATA
00266          .  IF(IDRWON.EQ.1)
00267          .  .  ADR=50+KSAV
00268          .  .  CALL DPW(IWLL,ADR,VAL,108.)
00269          .  ...FIN
00270    C RETRIVE FROM FILE O TEMPORARY FILE AND WRITE TO BINARY FILE
00271    C IF FILE NEEDED ROTATED
00272          .  IF(ISAVTP.EQ.1)
00273          .  .  IF(IROT.NE.0)
00274          .  .  .  DO(IY=1,NDSY)
00275          .  .  .  .  ADR=IY*4.
00276          .  .  .  .  CALL DPR(ITMPTR,ADR,A,NDSX*2.*NAQS)
00277          .  .  .  .  WRITE(5) (A(K),K=1,NDSX)
```

```
00278          .  .  .  .  IF(NAQS.GE.2) WRITE(4) (A(K),K=NDSX+1,NDSX*2)
00279          .  .  .  .  IF(NAQS.GE.3) WRITE(5) (A(K),K=NDSX*2+1,NDSX*3)
00280          .  .  .  ...FIN
00281          .  .  ...FIN
00282          .  .  CLOSE(UNIT=3)
00283          .  .  IF(NAQS.GE.2) CLOSE(UNIT=4)
00284          .  .  IF(NAQS.GE.3) CLOSE(UNIT=5)
00285          .  ...FIN
00286          ...FIN
00287    C NOW WRITE OUT THE HYDROGRAPH FILES HERE
00288          IF(IDRWDN.EQ.1)
00289          .  NWL=9216-7*NTPLSV
00290          .  NLEF=NWL/2
00291          .  IAPOS=7*NTPLSV
00292          .  IF(MOD(IAPOS,2).NE.0) IAPOS=IAPOS+1
00293          .  IAPOS=IAPOS/2+1
00294          .  NWL=NLEF/NTPLSV
00295          .  WHEN(NWL.GE.NWELLS)
00296          .  .  NPGS=1
00297          .  .  NWLPP=NWELLS
00298          .  ...FIN
00299          .  ELSE
00300          .  .  NPGS=NWELLS/NWLL
00301          .  .  IF(MOD(NWELLS,NWL).NE.0) NPGS=NPGS+1
00302          .  .  NWLPP=NWL
00303          .  ...FIN
00304          .  DO(IPG=1,NPGS)
00305          .  .  ISWL=(IPG-1)*NWLPP+1
00306          .  .  IEWL=IPG*NWLPP
00307          .  .  DO(ITP=1,NTPLSV)
00308          .  .  .  IPOS=(ITP-1)*NWLPP+IAPOS
00309          .  .  .  ADR=50+ITP
00310          .  .  .  CALL DPR(IWLL,ADR,VAL,108.)
00311          .  .  .  DO(IWL=ISWL,IEWL)
00312          .  .  .  .  I=(IWL-ISWL)+1+IPOS
00313          .  .  .  .  A(I)=VAL(IWL)
00314          .  .  .  ...FIN
00315          .  .  .  A(ITP)=ATIME
00316          .  .  .  IA(NTPLSV*2+(ITP-1)*5+1)=ITIMEA(1)
00317          .  .  .  IA(NTPLSV*2+(ITP-1)*5+2)=ITIMEA(2)
00318          .  .  .  IA(NTPLSV*2+(ITP-1)*5+3)=ITIMEA(3)
00319          .  .  .  IA(NTPLSV*2+(ITP-1)*5+4)=ITIMEA(4)
00320          .  .  .  IA(NTPLSV*2+(ITP-1)*5+5)=ITIMEA(5)
00321          .  .  ...FIN
00322          .  .  DO(IWL=ISWL,IEWL)
00323          .  .  .  IP=IWL-ISWL+1
00324          .  .  .  N=IWLAD(IWL)
00325          .  .  .  ADR=N-1
00326          .  .  .  CALL DPR(IWLL,ADR,WELNAM,32.)
00327    D     .  .  .  WRITE(7,*) IPG,IWL,ISWL,IEWL,IP,N,ADR
00328    D     .  .  .  WRITE(7,302) WELNAM,WELLID
00329    D302  .  .  .  FORMAT(7A4,10X,3A4)
00330          .  .  .  OPEN(UNIT=3,NAME=WELNAM,TYPE='NEW')
00331          .  .  .  VALM=A(IAPOS+IP)
00332          .  .  .  WRITE(3,500) WELLID,NTPLSV,WELLX,WELLY,IAQNUM,VALO,ITIME,RMIN(1)
00333    500   .  .  .  FORMAT(3A4,I3,2F11.2,I2,F11.2,I5,4I3,F10.0)
```

```
00334          .  .  .  DO(ITP=1,NTPLSV)
00335          .  .  .  .  IPOS=(ITP-1)*NWLPP+IAPOS
00336          .  .  .  .  IPOSS=IPOS+IP
00337          .  .  .  .  WRITE(3,501) A(ITP),A(IPOSS),(IA((ITP-1)*5+NTPLSV*2+I),I=1,5)
00338    501   .  .  .  .  FORMAT(2F12.2,5I5)
00339          .  .  .  ...FIN
00340          .  .  .  CLOSE(UNIT=3)
00341          .  .  ...FIN
00342          .  ...FIN
00343          ...FIN
00344          RETURN
00345          END
```

(FLECS VERSION 22.46)

```
00346          SUBROUTINE UNROT(A,NDSX,NDSY,NAQS,NPGS,ISTLN,ISTPLN,ITMPTR)
00347    C
00348          DIMENSION VAL(384),A(4608)
00349    C
00350          DO(ILNR=1,NDSY)
00351          .  ADR=ILNR*4.
00352          .  ILNRA=NDSY+1-ILNR
00353          .  CALL DPW(ITMPTR,ADR,VAL,NDSX*NAQS*2.)
00354          .  DO(ICLR=ISTLN,ISTPLN)
00355          .  .  ICLRA=ICLR-ISTLN+1
00356          .  .  DO(IAQ=1,NAQS)
00357          .  .  .  IXPOS=ICLR+(IAQ-1)*NDSX
00358          .  .  .  IPOS=(ICLRA-1)*NDSY*NAQS+(IAQ-1)*NDSY+ILNRA
00359          .  .  .  VAL(IXPOS)=A(IPOS)
00360          .  .  ...FIN
00361          .  ...FIN
00362          .  CALL DPW(ITMPTR,ADR,VAL,NDSX*NAQS*2.)
00363          ...FIN
00364          RETURN
00365          END
```

(FLECS VERSION 22.46)

```
00366          SUBROUTINE FGEN2(B,B1,IXSIZE,NX,NY,X,Y,PRED,IFLG)
00367    C     SUBROUTINE TO GENERATE A FUNCTION
00368    C     OF TWO VARIABLES LINEAR INTERPOLATION
00369          DIMENSION B(IXSIZE),B1(IXSIZE)
00370          IX=X
00371          IY=Y
00372          ILL=IX
00373          ILR=ILL+1
00374          DX=X-IX
00375          DY=Y-IY
00376          CALL INTERP(B(ILL),B(ILR),B1(ILL),B1(ILR)
00377         1,DX,DY,1.,PRED,IFLG)
00378          RETURN
00379          END
```

(FLECS VERSION 22.46)

```
      SUBROUTINE INTERP(P1,P2,P3,P4,XF,YF,DX,P,IFLG)
      IFLG=0
      P=0.
      IN=0
      IF(P1.NE.0) IN=IN+1
      IF(P2.NE.0) IN=IN+1
      IF(P3.NE.0) IN=IN+1
      IF(P4.NE.0) IN=IN+1
      IF(IN.LT.3) RETURN
      Z=XF/DX
      E=YF/DX
      Z1=-1.0*(1.0-Z)
      E1=-1.0*(1.0-E)
      IF(IN.EQ.3) GO TO 1
      P5=(P2-P1)*XF/DX+P1
      P6=(P4-P3)*XF/DX+P3
      P=(P6-P5)*YF/DX+P5
      IFLG=1
      RETURN
1     IF(P1.EQ.0.AND.Z+E.GE.1.) GO TO 10
      IF(P2.EQ.0.AND.Z-E.LE.0.) GO TO 20
      IF(P3.EQ.0.AND.Z-E.GE.0) GO TO 30
      IF(P4.EQ.0.AND.Z+E.LE.1) GO TO 40
      IFLG=0
      P=0.
      RETURN
10    P=P4*(1.0-Z1-E1)+P3*Z1+P2*E1
      IFLG=1
      RETURN
20    P=P3*(1.0-Z-E1)+P4*Z+P1*E1
      IFLG=1
      RETURN
30    P=P2*(1.-Z1-E)+P1*Z1+P4*E
      IFLG=1
      RETURN
40    P=P1*(1.0-Z-E)+P2*Z+P3*E
      IFLG=1
      RETURN
      END
```

(FLECS VERSION 22.46)

THIS PAGE INTENTIONALLY
LEFT BLANK

MXED

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DP1:MXED=MXED
[1,1]BNWLIB/LB
/
ASG=TI:6
ASG=DBO:2
/
```

```
      C       ***** MXED *****
      C FILEQ MATRIX EDITING PROGRAM FOR VTT PROGRAM
      C MXED.FTN
0001          DIMENSION BUFF(128),NTP(2,128),IMAP(12),NNO(256)
0002          COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,IBK(250)
0003          EQUIVALENCE (BUFF(1),NTP(1,1))
0004          DATA IMAP/1,2,3,4,5,5,7,8,9,10,11,12/
0005          WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0006          READ(6,*) IUN
0007          IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0008          WRITE(6,110)
0009          READ(6,103) IFLD
0010          IF(IFLD.EQ.0) CALL DPFILE(2,'VTTFIELDS',2401.,NEF)
0011          IF(IFLD.EQ.1) CALL DPFILE(2,'VTTSMALL',2401.,NEF)
0012          CALL DPR(NEF,0.,N,256.)
0013          LNY =1
0014          ICLX =1
0015          IARY =1
0016    1     WRITE(6,100)
0017          WRITE(6,101)
0018          WRITE(6,102)
0019          READ(6,103) IAA
0020          IF(IAA .EQ. 0 .OR. IAA .GT. 12) GO TO 5
0021          IA =IAA
0022          IARY =IMAP(IA)
0023    5     WRITE(6,104)
0024          READ(6,103) ILN,ICO
0025          IF(ILN .EQ. 0 .AND. ICO .EQ. 0) GO TO 10
0026          LNY =ILN
0027          ICLX =ICO
0028    10    CALL DSPY(LNY,ICLX,IA,NEF)
0029          WRITE(6,105)
0030          READ(6,103) INW
0031          IF(INW .EQ. 0) GO TO 25
0032          WRITE(6,106)
0033          IF(IARY .NE. 5) READ(6,107) VAL
0034          IF(IA .EQ. 5) READ(6,103) IVAL
0035          IF(IA .EQ. 6) READ(6,109) IVAL
0036          RB =(LNY-1)*12 +IARY
0037          CALL DPR(NEF,RB,BUFF,256.)
0038          IF(IARY .EQ. 5) GO TO 15
0039          BUFF(ICLX) =VAL
0040          GO TO 20
0041    15    J =IA -4
0042          NTP(J,ICLX) =IVAL
0043    20    CALL DPW(NEF,RB,BUFF,256.)
0044          CALL DSPY(LNY,ICLX,IA,NEF)
      C
      C                 NEW NODE NUMBERS
      C
0045          IF(IA .NE. 6) GO TO 25
0046          WRITE(6,111)
0047          READ(6,103) INM
0048          IF(INM .EQ. 0) GO TO 25
0049          NCNT =1
0050          DO 40 ILN =1,LINES
```

```
0051            RB =(ILN-1)*12. +5.
0052            CALL DPR(NEF,RB,NTP,2.*ICOL)
0053            CALL DPR(NEF,RB+1.,NNO,2.*ICOL)
0054            NACTV =0
0055            NSTR =1
0056            DO 32 I =1,128
0057     32     NNO(I) =0
0058            DO 30 I =1,ICOL
0059            NODE =NTP(2,I)
0060            NTP(1,I) =0
0061            IF(NODE .EQ. 1 .OR. NODE .EQ. 2) GO TO 30
0062            NCNT =NCNT +1
0063            NTP(1,I) =NCNT
0064     30     CONTINUE
0065            DO 35 I =1,ICOL
0066            NODE =NTP(2,I)
0067            IF(NODE .EQ. 1 .OR. NODE .EQ. 2) GO TO 35
0068            NACTV =NACTV +1
0069            NSTR =NSTR +1
0070            NNO(NSTR) =I
0071     35     CONTINUE
0072            NNO(1) =NACTV
0073            CALL DPW(NEF,RB,NTP,2.*ICOL)
0074            CALL DPW(NEF,RB+1.,NNO,2.*ICOL)
0075     40     CONTINUE
0076            N =NCNT
0077            CALL DPW(NEF,0.,N,256.)
0078     25     WRITE(6,108)
0079            READ(6,103) IX
0080            IF(IX .NE. 0) STOP
0081            GO TO 1
0082    100     FORMAT(1X,'SPECIFY ARRAY (CR =SAME)')
0083    101     FORMAT(1X,'H  BTM  K  Q  NNO  NTP  TOP  TRN  OCN  OLOH  STO  Q2')
0084    102     FORMAT(1X,'1   2   3  4   5    6    7    8    9    10    11
                112')
0085    103     FORMAT(2I6)
0086    104     FORMAT(1X,'LINE NO. , COLUMN NO. (2I4,CR=SAME)')
0087    105     FORMAT(1X,'CHANGE VALUE ? (CR =NO, 1 =YES)')
0088    106     FORMAT(1X,'NEW VALUE =')
0089    107     FORMAT(F10.0)
0090    108     FORMAT(1X,'EXIT ? (CR=NO,1=YES)')
0091    109     FORMAT(O10)
0092    110     FORMAT(1X,'LARGE OR SMALL FIELD ? :CR=LG,1=SM')
0093    111     FORMAT(1X,'RE-NUMBER NODES ? :1=YES,CR=NO')
0094            END
```

```
0001            SUBROUTINE DSPY(IY,IX,IA,NEF)
0002            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT
0003            DIMENSION BUFF(128),NTP(2,128)
0004            EQUIVALENCE (BUFF(1),NTP(1,1))
0005            IARY =IA
0006            IF(IA .EQ. 6) IARY =5
0007            IF(IA .EQ. 1) WRITE(6,100)
0008            IF(IA .EQ. 2) WRITE(6,101)
0009            IF(IA .EQ. 3) WRITE(6,102)
0010            IF(IA .EQ. 4) WRITE(6,103)
0011            IF(IA .EQ. 5) WRITE(6,104)
0012            IF(IA .EQ. 6) WRITE(6,105)
0013            IF(IA .EQ. 7) WRITE(6,106)
0014            IF(IA .EQ. 8) WRITE(6,107)
0015            IF(IA .EQ. 9) WRITE(6,113)
0016            IF(IA .EQ. 10) WRITE(6,114)
0017            IF(IA .EQ. 11) WRITE(6,115)
0018            IF(IA .EQ. 12) WRITE(6,116)
0019            DO 20 ILR =1,-1,-1
0020            ILN =IY +ILR
0021            IF(ILN .LT. 1 .OR. ILN .GT. LINES) GO TO 20
0022            RB =(ILN-1)*12 +IARY
0023            CALL DPR(NEF,RB,BUFF,256,)
0024            IF(IARY .EQ. 5) GO TO 10
0025            IF(IARY .NE. 11) GO TO 15
0026            WRITE(6,117) ILN,BUFF(IX-1),BUFF(IX),BUFF(IX+1)
0027            GO TO 20
0028      15    WRITE(6,108) ILN,BUFF(IX-1),BUFF(IX),BUFF(IX+1)
0029            GO TO 20
0030      10    J =IA -4
0031            IF(J.EQ.1)WRITE(6,109)ILN,NTP(1,IX-1),NTP(1,IX),NTP(1,IX+1)
0032            IF(J.EQ.2)WRITE(6,110)ILN,NTP(2,IX-1),NTP(2,IX),NTP(2,IX+1)
0033      20    CONTINUE
0034            IX1 =IX-1
0035            IX2 =IX+1
0036            WRITE(6,112) IX1,IX,IX2
0037            WRITE(6,111)
0038            RETURN
0039      100   FORMAT(/1X,'LINE NO.',20X,'POTENTIAL (FT)'/)
0040      101   FORMAT(/1X,'LINE NO.',20X,'BOTTOM ELV. (FT)'/)
0041      102   FORMAT(/1X,'LINE NO.',20X,'HYD. COND. (FT/DAY)'/)
0042      103   FORMAT(/1X,'LINE NO.',20X,'Q-SOURCE (CFD)'/)
0043      104   FORMAT(/1X,'LINE NO.',20X,'NODE NO.'/)
0044      105   FORMAT(/1X,'LINE NO.',20X,'NODE TYPE'/)
0045      106   FORMAT(/1X,'LINE NO.',20X,'TOP OF AQF (FT)'/)
0046      107   FORMAT(/1X,'LINE NO.',20X,'TRANS. COFF. (FT**2/DAY)'/)
0047      113   FORMAT(/1X,'LINE NO.',20X,'OCEAN TRANS. (FT**2/DAY)'/)
0048      114   FORMAT(/1X,'LINE NO.',20X,'OLD POTENTIAL (FT)'/)
0049      115   FORMAT(/1X,'LINE NO.',20X,'STORAGE COFF.'/)
0050      116   FORMAT(/1X,'LINE NO.',20X,'NEXT Q (CFD)'/)
0051      108   FORMAT(1X,I4,10X,3F12.1)
0052      109   FORMAT(1X,I4,10X,3I12)
0053      110   FORMAT(1X,I4,10X,3O12)
0054      111   FORMAT(35X,'COLUMN NO.'/)
0055      112   FORMAT(/15X,3I12)
0056      117   FORMAT(1X,I4,10X,3F12.6)
```

```
DB0:SUBGEN=SUBGEN,NTERP
[1,1]8NWLIB/LB
/
ASG=DB0:2,TI:6,LP:5
/
```

```
        C       ***** SUBGEN *****
        C       PROGRAM: SUBGEN
        C       FUNCTION: GENERATES ARRAYS AND LIST OF B.C. FOR SPECIFIED
        C                 SUBREGION (VTTSMALL)
        C      MODIFIED FOR MULTIAQUIFER SYSTEMS
        C       AUTHOR: H. P. FOOTE
        C
0001            BYTE STAT,DOIT(2),TITLE
0002            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,CONV,BIAS,NAQS,IOXFER
                1,IROT,MAXIT,DELX,IAUTO,OMEGA,NTMPLN,NOTIMS,ITROS,IWO,TIME,STIME
                2,ETIME,IYEAR,MONTH,IDAY,IHOUR,IMIN,IYEARL,MONTHL,IDAYL
                3,IHOURL,MINL,NOSX,NOSY,NAQFRS,NUCONF,DOIT,STOCF,TITLE(80)
                4,LEAK,SEALVL,DFSET,ITVN,ILGSML,ISMLSM,XLLC,YLLC,ISBLVL
0003            COMMON /HEADS/ IHD(256)
0004            REAL LEAK
0005            DIMENSION ARY(128,9),H2(128),H0(128),RCR(4),ACOFF(4,27),AA(4)
                1,FIELD(128,24),FDINTP(128,9),FSQR(9,9),FLDCR(4)
0006            INTEGER NTP(2,128,9),NTP2(2,128),NTP0(2,128),NTPCR(4),NDLST(2,9)
                1,NSQR(9,9),SMLIN,SMCOL,NNO(256),IATP(10),IINTP(10)
                2,LIST(2048),NDLSTA(16),INPUT(5)
0007            EQUIVALENCE (ARY(1,1),NTP),(FIELD(1,5),NTP0),(FIELD(1,17),NTP2)
                1,(FIELD(1,1),H0),(FIELD(1,13),H2),(IHD(87),SLEAK)
                2,(IHD(15),SDELX),(IHD(96),SXLLC),(ARY(1,1),LIST)
                3,(IHD(98),SYLLC),(IHD(100),ISUBLV),(IHD(101),MAG)
                4,(NDLST(1,1),NDLSTA)
0008            DATA ACOFF/ 1.,1.,1.,1.,0.,0.,0.,0.,0.,0.,0.,0.,.5,0.,.5,
                1).,0.,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.
                2,0.,1.,.5,0.,.5,1.,0.,0.,1.,0.,0.,.5,1.,0.,0.,1.,.5,0.,.5
                3,1.,0.,0.,1.,.5,0.,.5,1.,0.,0.,1.,.5,0.,0.,1.,0.,0.,.5,.5,0.
                4,0.,1.,1.,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.
                5,1.,1.,0.,.5,1.,1.,0.,1.,1.,.5,.5,1.,1.,0./
0009            DATA IATP/0,0,0,2,0,0,1,0,0,2/
0010            DATA IINTP/1,2,3,4,7,8,9,10,11,12/
0011            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0012            READ(6,*) IUN
0013            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
        C
0014            CALL OPFILE(2,'VTTSMALL',2401.,NES)
0015            CALL OPFILE(2,'VTTFIELDS',2401.,NEF)
0016            CALL OPFILE(2,'VTTLIST',624.,NEL)
0017            CALL OPR(NEF,0.,N,100.)
0018            CALL OPR(NEF,0.,IHD,256.)
0019    1       WRITE(6,102)
0020            WRITE(6,100)
0021            READ(6,101) INPUT
0022            IF(IROT .EQ. 1) GO TO 5
0023            IXCL =INPUT(1)
0024            IXCL2 =INPUT(2)
0025            IYLN =INPUT(3)
0026            IYLN2 =INPUT(4)
0027            IXNO=NOSX
0028            GO TO 6
0029    5       IXCL =NOSY -INPUT(4) +1
0030            IXCL2 =NOSY -INPUT(3) +1
0031            IYLN =INPUT(1)
0032            IYLN2 =INPUT(2)
```

```
0033              IXWD=NOSY
0034      6       MAG =INPUT(5)
0035              IWIDE =IXCL2 -IXCL
0036              IHI =IYLN2 -IYLN
0037              SMLIN0=IHI*MAG+1
0038              SMCOL0=IWIDE*MAG+1
0039              SMLIN=SMLIN0
0040              SMCOL=SMCOL0*NAQS
          D       WRITE(5,*) IXCL,IXCL2,IYLN,IYLN2,IWIDE,IHI,SMLIN0,SMCOL0
          D       1,SMLIN,SMCOL
0041              IF(MAG*IWIDE .GT. 126 .OR. MAG*IHI .GT. 200) GO TO 1
          C
          C                 INTERPOLATE NODE TYPES
          C
0042              DO 220 ILN =IYLN,IYLN2 -1
0043              RB =(ILN-1)*12 +1
          D       WRITE(5,*) ILN,RB
0044              CALL DPR(NEF,RB,H0,24,*256,)
0045              DO 10 IAQ=1,NAQS
0046              IXCL0=IXCL+(IAQ-1)*IXWD
0047              IXCL20=IXCL2+(IAQ-1)*IXWD
          D       WRITE(5,*) 'IAQ',IAQ,IXCL0,IXCL20
0048              DO 10 ICL =IXCL0,IXCL20 -1
0049              NTPCR(1) =NTP0(2,ICL)
0050              NTPCR(2) =NTP0(2,ICL+1)
0051              NTPCR(3) =NTP2(2,ICL+1)
0052              NTPCR(4) =NTP2(2,ICL)
          D       WRITE(5,*) 'ICL',ICL
0053              CALL NTERP(NTPCR,NSQR,MAG)
0054              DO 10 J =1,MAG+1
          D       WRITE(5,*) 'J',J
0055              DO 10 I =1,MAG+1
          D       WRITE(5,*) 'I',I
0056              IX=(ICL-IXCL0)*MAG+I+(IAQ-1)*IFIX(SMCOL0)
0057              NTP(2,IX,J) =NSQR(I,J)
          D       WRITE(5,*) 'IX',IX,NSQR(I,J)
0058      10      CONTINUE
0059              DO 15 J =1,MAG+1
0060              WB =((ILN-IYLN)*MAG +J -1)*12 +5
          D       WRITE(5,*) J,WB
0061              CALL DPW(NES,WB,NTP(1,1,J),256,)
0062      15      CONTINUE
          C
          C                 INTERPOLATION OF OTHER FIELDS
          C
0063              DO 220 IFL =1,10
          D       WRITE(5,*) 'IFL',IFL
0064              IFLD =IINTP(IFL)
0065              IARFG =IATP(IFL)
0066              IF(DDIT(2).NE.1.AND.IFL.EQ.6) IARFG=1
0067              DO 215 IAQ=1,NAQS
0068              IXCL0=IXCL+(IAQ-1)*IXWD
0069              IXCL20=IXCL2+(IAQ-1)*IXWD
          D       WRITE(5,*) 'IAQ',IAQ,IXCL0,IXCL20
0070              DO 215 ICL =IXCL0,IXCL20-1
0071              FLDCR(1) =FIFLD(ICL,IFLD)
```

```
0072            FLDCR(2) =FIELD(ICL+1,IFLD)
0073            FLDCR(3) =FIELD(ICL+1,IFLD+12)
0074            FLDCR(4) =FIELD(ICL,IFLD+12)
0075            IXCR =(ICL -IXCL0)*MAG+(IAQ-1)*IFIX(SMCOL0)
        D       WRITE(5,*) 'ICL',ICL,IXCR
0076            CALL FINTP(FLDCR,FSQR,MAG,IARFG,NTP,IXCR)
0077            DO 215 J =1,MAG+1
0078            DO 215 I =1,MAG+1
0079            IX =(ICL -IXCL0)*MAG +I+(IAQ-1)*IFIX(SMCOL0)
        D       WRITE(5,*) 'I,J,IX',I,J,IX
0080            FDINTP(IX,J) =FSQR(I,J)
0081    215     CONTINUE
0082            DO 220 J =1,MAG+1
0083            WB =((ILN-IYLN)*MAG +J -1)*12. +IFLD
        D       WRITE(5,*) 'J,WB',J,WB
0084            CALL DPW(NES,WB,FDINTP(1,J),256.)
0085    220     CONTINUE
        C
        C               DEFINE NEW BOUNDRY NODES AND NUMBERS
        C
        C       SMLIN =IHI*MAG +1
        C       SMCOL =IWIDE*MAG +1
0086            NCNT =0
0087            CALL DPR(NES,5.,NTP(1,1,2),2.*SMCOL)
0088            CALL DPR(NES,17.,NTP(1,1,3),2.*SMCOL)
0089            DO 50 ILN =1,SMLIN
0090            DO 30 IAQ=1,NAQS
0091            IADR=(IAQ-1)*SMCOL0
0092            DO 30 ICLG =1,SMCOL0
0093            ICL=ICLG+IADR
0094            NODE =NTP(2,ICL,2)
0095            IF(NODE .EQ. 1 .OR. NODE .EQ. 2) GO TO 30
0096            NTPCR(1) =NTP(2,ICL+1,2)
0097            IF(ICLG .EQ. SMCOL0) NTPCR(1) =1
0098            NTPCR(2) =NTP(2,ICL,3)
0099            IF(ILN .EQ.SMLIN) NTPCR(2) =1
0100            NTPCR(3) =1
0101            IF(ICLG .EQ. 1) GO TO 16
0102            NTPCR(3) =NTP(2,ICL-1,2)
0103    16      NTPCR(4) =NTP(2,ICL,1)
0104            IF(ILN .EQ. 1) NTPCR(4) =1
0105            IF(NODE .GT. 2) NODE =NODE-5
0106            DO 17 I =1,4
0107    17      AA(I) =ACOFF(I,NODE+1)
0108            DO 18 I =1,4
0109            IF(NTPCR(I) .EQ. 1 .AND. AA(I) .GT. 0.) GO TO 20
0110    18      CONTINUE
0111            GO TO 30
0112    20      NTP(2,ICL,2) =2
0113    30      CONTINUE
0114            DO 35 ICL =1,SMCOL
0115            NODE =NTP(2,ICL,2)
0116            NTP(1,ICL,2) =0
0117            IF(NODE .EQ. 1 .OR. NODE .EQ. 2) GO TO 35
0118            NCNT =NCNT +1
0119            NTP(1,ICL,2) =NCNT
```

```
0120    35      CONTINUE
0121            WB =(ILN-1)*12. +5.
0122            CALL DPW(NES,WB,NTP(1,1,2),2.*SMCOL)
0123            IF(ILN .EQ. SMLIN) GO TO 50
0124            DO 40 J =2,3
0125            DO 40 I =1,SMCOL
0126            ARY(I,J-1) =ARY(I,J)
0127    40      CONTINUE
0128            RB =(ILN+1)*12. +5.
0129            CALL DPR(NES,RB,NTP(1,1,3),2.*SMCOL)
0130    50      CONTINUE
        C
        C       GENERATE ACTIVE NODE NUMBER ARRAY
        C
0131            DO 60 ILN =1,SMLIN
0132            RB =(ILN-1)*12. +5.
0133            CALL DPR(NES,RB,NTP0,2.*SMCOL)
0134            NACTV =0
0135            NSTR =1
0136            DO 52 I =1,128
0137    52      NNO(I) =0
0138            DO 55 ICL =1,SMCOL
0139            NODE =NTP0(2,ICL)
0140            IF(NODE .EQ. 1 .OR. NODE .EQ. 2) GO TO 55
0141            NACTV =NACTV +1
0142            NSTR =NSTR +1
0143            NNO(NSTR) =ICL
0144    55      CONTINUE
0145            NNO(1) =NACTV
0146            CALL DPW(NES,RB+1.,NNO,128.)
0147    60      CONTINUE
        C
        C               HALF BAND CALCULATION
        C
0148            IMAX =0
0149            DO 70 ILN =1,SMLIN-1
0150            RB =(ILN-1)*12. +5.
0151            CALL DPR(NES,RB,NTP0,256.)
0152            CALL DPR(NES,RB+12.,NTP2,256.)
0153            DO 70 IP =1,SMCOL
0154            IN0 =NTP0(1,IP)
0155            IN1 =NTP2(1,IP)
0156            IF(IN0 .EQ. 0 .OR. IN1 .EQ. 0) GO TO 70
0157            IDF =IN1 -IN0
0158            IF(IDF .GT. IMAX) IMAX =IDF
0159    70      CONTINUE
        C
        C               NODE LIST GENERATION
        C
0160            ISMW0=SMCOL0
0161            NPAIR =0
0162            NHI =(IYLN2 -IYLN)*MAG +1
0163            NW0 =((IXCL2 -IXCL)*MAG +1)*NAQS
0164            LIST(2) =MAG
0165            DO 300 IPR =1,IWIDE*NAQS
0166            IX0 =IXCL +IPR -1
```

```
                ISADD=0
                ILADD=0
                IF(IPR.LE.2*IWIDE) GO TO 7777
                IX0=IX0+2*IXWD-2*IWIDE
                ISADD=ISADD+2*ISMWD
                ILADD=ILADD+2*IXWD
                GO TO 7776
        7777    CONTINUE
                IF(IPR.LE.IWIDE) GO TO 7776
                IX0=IX0+IXWD-IWIDE
                ISADD=ISADD+ISMWD
                ILADD=ILADD+IXWD
        7776    CONTINUE
        D       WRITE(5,*) 'IPR,IWIDE,IX0,ISADD,ILADD'
        D       WRITE(5,*) IPR,IWIDE,IX0,ISADD,ILADD
                NPAIR =NPAIR +1
                IXCLL=IXCL+ILADD
                CALL STOLST(LIST,NPAIR,IX0,IYLN,IX0+1,IYLN,IXCLL,IYLN,MAG,ISADD)
        300     CONTINUE
        D       WRITE(5,898) LIST
                DO 305 IPR =1,IH1
                IY0 =IYLN+IPR-1
                NPAIR =NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL,IY0,IXCL,IY0+1,IXCL,IYLN,MAG,0)
                NPAIR =NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL2,IY0,IXCL2,IY0+1,IXCL,IYLN,MAG,0)
                IF(NAGS.LE.1) GO TO 7775
                NPAIR=NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL+IXWD,IY0,IXCL+IXWD,IY0+1,
               1IXCL+IXWD,IYLN,MAG,ISMWD)
                NPAIR=NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL2+IXWD,IY0,IXCL2+IXWD,IY0+1,
               1IXCL+IXWD,IYLN,MAG,ISMWD)
        7775    CONTINUE
                IF(NAGS.LE.2) GO TO 7774
                NPAIR=NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL+IXWD*2,IY0,IXCL+IXWD*2,IY0+1,
               1IXCL+IXWD*2,IYLN,MAG,ISMWD*2)
                NPAIR=NPAIR+1
                CALL STOLST(LIST,NPAIR,IXCL2+IXWD*2,IY0,IXCL2+IXWD*2,IY0+1,
               1IXCL+IXWD*2,IYLN,MAG,ISMWD*2)
        7774    CONTINUE
        305     CONTINUE
        D       WRITE(5,898) LIST
                DO 310 IPR =1,IWIDE*NAGS
                IX0 =IXCL +IPR -1
                ISADD=0
                ILADD=0
                IF(IPR.LE.2*IWIDE) GO TO 8777
                IX0=IX0+2*IXWD-2*IWIDE
                ISADD=ISADD+2*ISMWD
                ILADD=ILADD+2*IXWD
                GO TO 8776
        8777    CONTINUE
                IF(IPR.LE.IWIDE) GO TO 8776
                IX0=IX0+IXWD-IWIDE
```

```
0215                ISADD=ISADD+ISHWD
0216                ILADD=ILADD+IXWD
0217      8776      CONTINUE
          D         WRITE(5,*) IPR,IWIDE,IX0,ISADD,ILADD
0218                NPAIR =NPAIR +1
0219                IXCLL=IXCL+ILADD
0220                CALL STOLST(
                    1LIST,NPAIR,IX0,IYLN2,IX0+1,IYLN2,IXCLL,IYLN,MAG,ISADD)
0221      310       CONTINUE
          D         WRITE(5,898) LIST
0222                LIST(1) =NPAIR
          C
          C                 SET LIST ENTRIES
          C
0223                DO 330 IPR =1,NPAIR
0224                INDX =(IPR-1)*(2*(MAG+1)+5)+7
          D         WRITE(5,*) 'I,INDX',I,INDX
0225                DO 315 I =1,2*(MAG+1)
0226                IND1 =INDX +I
0227                NDLSTA(I) =LIST(IND1)
          D         WRITE(5,*) 'I,IND1,INDX,LIST(IND1)',I,IND1,INDX,LIST(IND1)
0228      315       CONTINUE
0229                IPAR =MAG+1
0230                IYLL =NDLST(2,1)
0231                RB =(IYLL-1)*12. +5.
0232                CALL DPR(NES,RB,NTP0,2,*SMCOL)
          D         WRITE(5,*) 'IPAR,IYLL,RB',IPAR,IYLL,RB
0233                DO 320 I =1,MAG+1
0234                IXN =NDLST(1,I)
0235                IYN =NDLST(2,I)
0236                IF(IYN .NE. IYLL)CALL DPR(NES,(IYN-1)*12.+5.,NTP0,2,*SMCOL)
          D         WRITE(5,*) 'I,IXN,IYN',I,IXN,IYN
0237                IF(NTP0(2,IXN) .EQ. 2) GO TO 320
0238                NDLST(1,I) =0
0239                NDLST(2,I) =0
0240                IPAR =IPAR-1
0241      320       CONTINUE
0242                LIST(INDX) =IPAR
0243                DO 330 I =1,2*(MAG+1)
0244                IND1 =INDX +I
0245                LIST(IND1) =NDLSTA(I)
          D         WRITE(5,*) 'I,IND1,LIST(IND1)',I,IND1,LIST(IND1)
0246      330       CONTINUE
          C
          D         WRITE(5,898) LIST
          C                 COMPRESS LIST
          C
0247                NWPR =0
0248                IWRDX =2
0249                NWRT =2*(MAG+1) +5
0250                DO 340 IPR =1,NPAIR
0251                IND =(IPR-1)*(2*(MAG+1)+5)+7
          D         WRITE(5,*) 'IPR,IND',IPR,IND
0252                IF(LIST(IND) .EQ. 0) GO TO 340
0253                DO 335 IC =1,NWRT
0254                IPTR =IWRDX +IC
```

```
0255              IPTR =IND -5 +IC
0256              LIST(IPTW) =LIST(IPTR)
        D         WRITE(5,*) 'IC,NWRT,IPTW,IPTR,LIST(IPTW)',IC,NWRT,IPTW,IPTR
        D         1,LIST(IPTW)
0257    335       CONTINUE
0258              NWPR =NWPR +1
0259              IWRDX =IWRDX +NWRT
        D         WRITE(5,*) 'NWPR,IWRDX',NWPR,IWRDX
0260    340       CONTINUE
0261              LIST(1) =NWPR
        D         WRITE(5,898) LIST
        D898      FORMAT(1X,25I5)
0262              CALL DPW(NEL,0.,LIST,2048.)
        C
        C                 FILL IN HEADER BLOCK
        C
0263              IROT =IHD(13)
0264              IHD(40) =SMCOL/NAQS
0265              IHD(41) =SMLIN
0266              IF(IROT .EQ. 1) IHD(40) =SMLIN
0267              IF(IROT .EQ. 1) IHD(41) =SMCOL/NAQS
0268              NTOT =NCNT
0269              SLEAK =LEAK/(MAG*MAG)
0270              SDELX =DELX/MAG
0271              IHD(94) =1
0272              IHD(93) =1
0273              IHD(95) =0
0274              ISUBLV =ISBLVL +1
0275              IF(IROT .EQ. 1) GO TO 75
0276              SXLLC =XLLC +(IXCL-1)*DELX
0277              SYLLC =YLLC +(IYLN-1)*DELX
0278              GO TO 76
0279    75        SXLLC =XLLC +(IYLN-1)*DELX
0280              SYLLC =YLLC +(ICOL-IXCL2)*DELX
0281    76        IHD(1) =NTOT
0282              IHD(2) =IMAX
0283              IHD(3) =SMLIN
0284              IHD(4) =SMCOL
0285              WRITE(4,*) 'UNK,M,NLINES,NCOLS,AQS'
0286              WRITE(6,*) NTOT,IMAX,SMLIN,SMCOL,NAQS
0287              CALL DPW(NES,0.,IHD,256.)
0288              STOP
0289    100       FORMAT(1X,'X0-COLL ,X1-COLL ,Y0-LINE ,Y1-LINE ,MAG :5I3')
0290    101       FORMAT(5I5)
0291    102       FORMAT(1X,'INPUT UNROTATED COORDINATES OF SUBREGION')
0292              END
```

```
FORTRAN IV-PLUS V02-510          16:49:54     30-MAY-79
SUBGEN.FTN       /TR:BLOCKS/WR

0001             SUBROUTINE STOLST(LIST,IPR,IX0,IY0,IX1,IY1,IXCL,IYLN,MAG,ISADD)
0002             DIMENSION LIST(2048)
0003             INDX =(IPR-1)*(2*(MAG+1)+5)+3
0004             LIST(INDX) =IX0
0005             LIST(INDX+1) =IY0
0006             LIST(INDX+2) =IX1
0007             LIST(INDX+3) =IY1
0008             LIST(INDX+4) =MAG+1
0009             IXINC =0
0010             IYINC =0
0011             IF(IX0 .EQ. IX1) IYINC =1
0012             IF(IY0 .EQ. IY1) IXINC =1
0013             IXSM =MAG*(IX0 -IXCL) +1+ISADD
0014             IYSM =MAG*(IY0 -IYLN) +1
0015             DO 5 IP =1,MAG+1
0016             IND1 =INDX +4 +2*IP-1
0017             LIST(IND1) =IXSM +(IP-1)*IXINC
0018             LIST(IND1+1) =IYSM +(IP-1)*IYINC
0019     5       CONTINUE
0020             RETURN
0021             END
```

```
0001            SUBROUTINE FINTP(FLDCR,FSQR,MAG,IARFG,NTP,IXCR)
0002            DIMENSION FLDCR(4),FSQR(9,9),AREA(4)
0003            INTEGER NTP(2,128,9),NTPCR(9,9),NCR(4)
0004            DATA AREA/1.,.5,.375,.625/
      C
0005            RMAG =MAG
0006            DO 5 J =1,MAG+1
0007            DO 5 I =1,MAG+1
0008            IX =IXCR +I
0009            NTPCR(I,J) =NTP(2,IX,J)
0010            FSQR(I,J) =0.
0011      5     CONTINUE
0012            NCR(1) =NTPCR(1,1)
0013            NCR(2) =NTPCR(MAG+1,1)
0014            NCR(3) =NTPCR(MAG+1,MAG+1)
0015            NCR(4) =NTPCR(1,MAG+1)
0016            NGO =0
0017            DO 10 I =1,4
0018            IF(NCR(I) .NE. 1) NGO =NGO +1
0019     10     CONTINUE
0020            IF(NGO .EQ. 0) RETURN
0021            GO TO(30,30,15,30),NGO
0022     15     DO 20 ICR =1,4
0023            IF(NCR(ICR) .EQ. 1) GO TO 25
0024     20     CONTINUE
0025            STOP
0026     25     ICA =ICR+1
0027            IF(ICA .EQ. 5) ICA =1
0028            ICB =ICR -1
0029            IF(ICB .EQ. 0) ICB =4
0030            ICC =ICR +2
0031            IF(ICC .GT. 4) ICC =ICC-4
0032            FLDCR(ICR) =FLDCR(ICA) +FLDCR(ICB) -FLDCR(ICC)
0033     30     A =FLDCR(2) -FLDCR(1)
0034            B =FLDCR(4) -FLDCR(1)
0035            C =FLDCR(3) +FLDCR(1) -FLDCR(4) -FLDCR(2)
0036            DO 35 J =1,MAG+1
0037            DO 35 I =1,MAG+1
0038            X =(I-1)/RMAG
0039            Y =(J-1)/RMAG
0040            F =FLDCR(1) +A*X +B*Y +X*Y*C
0041            IF(NTPCR(I,J) .EQ. 1) F =0.
0042            IF(IARFG .GT. 0) F =F/(RMAG*RMAG)
0043            FSQR(I,J) =F
0044     35     CONTINUE
0045            RETURN
0046            END
```

```
      C       *****SUBROUTINE NTERP*****
0001          SUBROUTINE NTERP(NTPCR,NSQR,MAG)
0002          DIMENSION ACOFF(4,27),CRAA(4,4)
0003          INTEGER NTPCR(4),NSQR(9,9),NCAT(7),IBTP(4),ISTEP(4,4)
              1,ITRI(3,4),IROT(4,4),ISDTP(4)
0004          DATA ACOFF/ 1.,1.,1.,1.,0.,0.,0.,0.,0.,0.,0.,0.,.5,0.,.5,
              11.,0.,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.
              2,0.,1.,.5,0.,.5,1.,0.,0.,1.,0.,0.,.5,1.,0.,0.,1.,.5,0.,.5
              3,1.,0.,0.,1.,.5,0.,.5,1.,0.,0.,1.,.5,0.,0.,1.,0.,0.,.5,.5,0.
              4,0.,1.,1.,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.,1.,1.,0.,.5,1.,.5,0.
              5,1.,1.,0.,.5,1.,1.,0.,1.,1.,.5,.5,1.,1.,0./
0005          DATA IBTP/12,10,8,14/
0006          DATA ISTEP/0,0,1,0,1,0,0,1,1,1,-1,0,0,1,0,-1/
0007          DATA ITRI/13,1,-1,11,1,1,9,-1,1,15,-1,-1/
0008          DATA IROT/0,0,1,1,1,0,-1,1,1,1,-1,-1,0,1,1,-1/
0009          DATA ISDTP/8,14,12,10/
      C
0010          DO 2 I =1,7
0011    2     NCAT(I) =0
0012          DO 10 I =1,4
0013          ITP =NTPCR(I)
0014          ITPA =ITP
0015          IF(ITP .LT. 3) GO TO 5
0016          ITPA =ITP/8 +3
0017          IF(ITPA .GE. 5) GO TO 5
0018          IRM =MOD(ITP,2)
0019          ITPA =IRM +3
0020    5     NCAT(ITPA+1) =NCAT(ITPA+1) +1
0021    10    CONTINUE
0022          DO 15 I =1,MAG+1
0023          DO 15 J =1,MAG+1
0024    15    NSQR(I,J) =0
0025          IF(NCAT(1) .EQ. 4) RETURN
0026          IF(NCAT(2) .EQ. 4) GO TO 20
0027          IF(NCAT(3) .EQ. 4) GO TO 20
0028          GO TO 30
0029    20    DO 25 I =1,MAG+1
0030          DO 25 J =1,MAG+1
0031          NSQR(I,J) =ITP
0032    25    CONTINUE
0033          RETURN
0034    30    NSQR(1,1) =NTPCR(1)
0035          NSQR(MAG+1,1) =NTPCR(2)
0036          NSQR(MAG+1,MAG+1) =NTPCR(3)
0037          NSQR(1,MAG+1) =NTPCR(4)
0038          IF(MAG .EQ. 1) RETURN
0039          DO 35 ICR =1,4
0040          NDTP =NTPCR(ICR)
0041          IF(NDTP .GT. 2) NDTP =NDTP-5
0042          DO 35 IDR =1,4
0043          CRAA(ICR,IDR) =ACOFF(IDR,NDTP+1)
0044    35    CONTINUE
0045          IG =NCAT(2) +1
0046          GO TO(40,100,200,200),IG
      C
      C               SQUARE CASE
```

```
      C
0047  40      DO 50 ICR =1,4
0048          IC1 =ICR +1
0049          IF(IC1 .EQ. 5) IC1 =1
0050          NODE =0
0051          IF(CRAA(ICR,ICR) .EQ. .5) NODE =IBTP(ICR)
0052          IF(NTPCR(ICR) .EQ. 2 .AND. NTPCR(IC1) .EQ. 2) NODE =2
0053          IX0 =ISTEP(1,ICR)*MAG +1
0054          IY0 =ISTEP(2,ICR)*MAG +1
0055          IDX =ISTEP(3,ICR)
0056          IDY =ISTEP(4,ICR)
0057          DO 50 IS =1,MAG-1
0058          IX =IS*IDX +IX0
0059          IY =IS*IDY +IY0
0060          NSQR(IX,IY) =NODE
0061  50      CONTINUE
0062          RETURN
      C
      C               TRIANGLE CASE
      C
0063  100     DO 105 ITP =1,4
0064          IF(NTPCR(ITP) .EQ. 1) GO TO 110
0065  105     CONTINUE
0066          STOP
0067  110     ICAS =ITP
0068          NODE =0
0069          IF(NCAT(3) .EQ. 3) NODE =2
0070          DO 120 J =1,MAG+1
0071          DO 120 I =1,MAG+1
0072          IX0 =IROT(1,ICAS)*MAG+1
0073          IY0 =IROT(2,ICAS)*MAG+1
0074          IX =(I-1)*IROT(3,ICAS) +IX0
0075          IY =(J-1)*IROT(4,ICAS) +IY0
0076          ND =NODE
0077          IF(I .LE. MAG-J+1) ND =1
0078          NSQR(IX,IY) =ND
0079  120     CONTINUE
0080          NSQR(1,1) =NTPCR(1)
0081          NSQR(MAG+1,1) =NTPCR(2)
0082          NSQR(MAG+1,MAG+1)=NTPCR(3)
0083          NSQR(1,MAG+1) =NTPCR(4)
0084          IF(NCAT(3) .EQ. 3) RETURN
0085          DO 130 ICR =ICAS,ICAS+1
0086          NCR =ICR +1
0087          IF(NCR .GT. 4) NCR =NCR-4
0088          NCR1 =ICR +2
0089          IF(NCR1 .GT. 4) NCR1 =NCR1-4
0090          NODE =0
0091          IF(CRAA(NCR,NCR) .EQ. .5) NODE =IBTP(NCR)
0092          IF(NTPCR(NCR) .EQ. 2 .AND. NTPCR(NCR1) .EQ. 2) NODE=2
0093          IX0 =ISTEP(1,NCR)*MAG +1
0094          IY0 =ISTEP(2,NCR)*MAG +1
0095          IDX =ISTEP(3,NCR)
0096          IDY =ISTEP(4,NCR)
0097          DO 130 IS =1,MAG-1
0098          IX =IS*IDX +IX0
```

```
0099              IY =IS*IDY +IY0
0100              NSQR(IX,IY) =NODE
0101     130      CONTINUE
0102              NODE =ITRI(1,ICAS)
0103              ICRA =ICAS +1
0104              IF(ICRA .GT. 4) ICRA =ICRA-4
0105              IF(NTPCR(ICRA) .EQ. 2 .AND. NTPCR(NCR1) .EQ. 2) NODE=2
0106              IDX =ITRI(2,ICAS)
0107              IDY =ITRI(3,ICAS)
0108              IX0 =ISTEP(1,NCR1)*MAG+1
0109              IY0 =ISTEP(2,NCR1)*MAG +1
0110              DO 140 IS =1,MAG-1
0111              IX =IS*IDX +IX0
0112              IY =IS*IDY +IY0
0113              NSQR(IX,IY) =NODE
0114     140      CONTINUE
0115              RETURN
          C
          C                 ONE POINT CASE
          C
0116     200      DO 205 I =1,MAG+1
0117              DO 205 J =1,MAG+1
0118              NSQR(I,J) =1
0119     205      CONTINUE
0120              NSQR(1,1) =NTPCR(1)
0121              NSQR(MAG+1,1) =NTPCR(2)
0122              NSQR(MAG+1,MAG+1) =NTPCR(3)
0123              NSQR(1,MAG+1) =NTPCR(4)
0124              IF(IG .EQ. 4) RETURN
          C
          C                 ONE SIDE CASE
          C
0125              IF(NTPCR(1) .EQ. 1) GO TO 210
0126              ICS =1
0127              IF(NTPCR(2) .EQ. 1) ICS =4
0128              GO TO 215
0129     210      ICS =2
0130              IF(NTPCR(2) .EQ. 1) ICS =3
0131     215      ICS1 =ICS +1
0132              IF(ICS1 .EQ. 5) ICS1 =1
0133              NODE =0
0134              IF(CRAA(ICS,ICS) .EQ. .5) NODE =ISDTP(ICS)
0135              IF(NTPCR(ICS) .EQ. 2 .AND. NTPCR(ICS1) .EQ. 2) NODE=2
0136              IX0 =ISTEP(1,ICS)*MAG +1
0137              IY0 =ISTEP(2,ICS)*MAG +1
0138              IDX =ISTEP(3,ICS)
0139              IDY =ISTEP(4,ICS)
0140              DO 220 IS =1,MAG-1
0141              IX =IS*IDX +IX0
0142              IY =IS*IDY +IY0
0143              NSQR(IX,IY) =NODE
0144     220      CONTINUE
0145              RETURN
0146              END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

BCTRN

THIS PAGE INTENTIONALLY
LEFT BLANK

```
        C       ***** BCTRN *****
        C       PROGRAM: BCTRN
        C       FUNCTION: INTERPOLATE POT. FOR SMALL REG. ,TRANSFER SMALL TO
        C                 LARGE FIELD ,SET ,LIST WELL QS, SET LEAKAGE COFF.
        C       AUTHOR: H. P. FOOTE
        C
0001            BYTE DOIT(2),TITLE,FILE(28)
0002            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,CONV,BIAS,NAQS,IOXFER
                1,IROT,MAXIT,DELX,IAUTO,OMEGA,NTMPLN,NOTIMS,ITROS,IWD,TIME,STIME
                2,ETIME,IYEAR,MONTH,IDAY,IHOUR,IMIN,IYEARL,MONTHL,IDAYL
                3,IHOURL,MINL,NOSX,NOSY,NAOFRS,NUCONF,DOIT,STOCF,TITLE(80)
                4,LEAK,SEALVL,OFSET,ITVN,ILGSML,ISMLSM,XLLC,YLLC,ISBLVL,MAG,ISPC
0003            REAL LEAK,DELXS,XLLCS,YLLCS
0004            DIMENSION NDLST(2,9),NDLSTL(18),H0(128),WELQS(3,25),H2(128)
                1,TBUF(128,12),FDINTP(128,12),FSQR(9,9),IWLAQ(25)
                2,FLOCR(4)
0005            INTEGER NTP(2,128,9),HDSM(128),INPUT(5),SMLN,SMCOL
                1,NTPSM(2,128)
0006            EQUIVALENCE (NDLST(1,1),NDLSTL),(TBUF(1,1),H0),(TBUF,FDINTP)
                1,(HDSM(15),DELXS),(HDSM(96),XLLCS),(HDSM(98),YLLCS)
                2,(HDSM(3),SMLN),(HDSM(4),SMCOL),(HDSM(40),NOSXS)
                3,(HDSM(41),NOSYS)
0007            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0008            READ(6,*) IUN
0009            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0010            CALL DPFILE(2,'VTTSMALL',2401.,NES)
0011            CALL DPFILE(2,'VTTFIELDS',2401.,NEF)
0012            CALL DPR(NEF,0.,N,100.)
0013            CALL DPR(NES,0.,HDSM,100.)
0014            MAGR =DELX/DELXS
0015            IHI =(NOSYS-1)/MAGR
0016            IWIDE =(NOSXS-1)/MAGR
0017            INPUT(1) =(XLLCS -XLLC)/DELX +1
0018            INPUT(2) =INPUT(1) +IWIDE
0019            INPUT(3) =(YLLCS -YLLC)/DELX +1
0020            INPUT(4) =INPUT(3) +IHI
0021            INPUT(5) =MAGR
0022    1       WRITE(6,100)
0023            READ(6,101) IM
0024            IF(IM .EQ. 1) GO TO 30
0025            WRITE(6,102)
0026            READ(6,101) IT
0027            IF(IT .EQ. 1) GO TO 2
0028            WRITE(6,108)
0029            READ(6,101) IWQ
0030            IF(IWQ .EQ. 1) GO TO 200
0031    3       WRITE(6,111)
0032            READ(6,101) ILK
0033            IF(ILK .EQ. 1) GO TO 300
0034            STOP
        C
        C       INTERPOLATE  POTENTIAL FROM LARGE REGION
        C
0035    2       IF(IROT .EQ. 1) GO TO 5
0036            IXCL =INPUT(1)
0037            IXCL2 =INPUT(2)
```

```
0038            IYLN =INPUT(3)
0039            IYLN2 =INPUT(4)
0040            IXWD=NOSX
0041            GO TO 6
0042     5      IXCL =NOSY -INPUT(4) +1
0043            IXCL2 =NOSY -INPUT(3) +1
0044            IYLN =INPUT(1)
0045            IYLN2 =INPUT(2)
0046            IXWD=NOSY
0047     6      MAG =INPUT(5)
0048            IWIDE =IXCL2 -IXCL
0049            IHI =IYLN2 -IYLN
0050            SMLIN0=IHI*MAG+1
0051            SMCOL0=IWIDE*MAG+1
0052            SMLIN=SMLIN0
0053            SMCOL=SMCOL0*NAQS
        C
        C               SET UP NODE TYPE ARRAY FROM SMALL REGION
        C
0054            DO 20 ILN =IYLN,IYLN2 -1
0055            RB =(ILN-1)*12 +1
0056            CALL DPR(NEF,RB,H0,256.)
0057            CALL DPR(NEF,RB+12.,H2,256.)
0058            DO 10 JJ =1,MAG+1
0059            ILS =(ILN -IYLN)*MAG +JJ
0060            RBS =(ILS-1)*12. +5.
0061            CALL DPR(NES,RBS,NTPSM,256.)
0062            DO 10 IX =1,SMCOL
0063            NTP(2,IX,JJ) =NTPSM(2,IX)
0064    10      CONTINUE
0065            IFLD =1
0066            IARFG=0
0067            DO 15 IAQ=1,NAQS
0068            IXCL0=IXCL+(IAQ-1)*IXWD
0069            IXCL20=IXCL2+(IAQ-1)*IXWD
0070            DO 15 ICL=IXCL0,IXCL20-1
0071            FLDCR(1) =H0(ICL)
0072            FLDCR(2) =H0(ICL+1)
0073            FLDCR(3) =H2(ICL+1)
0074            FLDCR(4) =H2(ICL)
0075            IXCR=(ICL-IXCL0)*MAG+(IAQ-1)*IFIX(SMCOL0)
0076            CALL FINTP(FLDCR,FSQR,MAG,IARFG,NTP,IXCR)
0077            DO 15 J =1,MAG+1
0078            DO 15 I =1,MAG+1
0079            IX=(ICL-IXCL0)*MAG+I+(IAQ-1)*IFIX(SMCOL0)
0080            FDINTP(IX,J) =FSQR(I,J)
0081    15      CONTINUE
0082            DO 20 J =1,MAG+1
0083            WB =((ILN-IYLN)*MAG +J -1)*12. +IFLD
0084            CALL DPW(NES,WB,FDINTP(1,J),256.)
0085    20      CONTINUE
0086            STOP
        C
        C               TRANSFER SMALL FIELD TO LARGE FIELD
        C
0087    30      CALL DPR(NES,0.,N,102.)
```

```
0088            ILGSML =0
0089            CALL DPW(NEF,0.,N,102.)
0090            DO 35 ILN =1,LINES
0091            RB =(ILN-1)*12. +1.
0092            CALL DPR(NES,RB,TBUF,12.*256.)
0093            CALL DPW(NEF,RB,TBUF,12.*256.)
0094     35     CONTINUE
0095            STOP
        C
        C               READ WELL LIST
        C
0096     200    WRITE(6,*) 'ENTER THE NAME OF THE WELL LIST FILE'
0097            READ(6,201) FILE
0098     201    FORMAT(20A1)
0099            N=ICHR(FILE)
0100            CALL ASNLUN(3,'DP',0,ISTAT)
0101            OPEN(UNIT=3,NAME=FILE,TYPE='OLD',READONLY)
0102            READ(3,101) NWELLS
0103            IF(NWELLS .EQ. 0) GO TO 1
0104            IF(NWELLS .GT. 25) NWELLS =25
0105            DO 205 J =1,NWELLS
0106            READ(3,1033) IWLAQ(J),WELQS(1,J),WELQS(2,J),WELQS(3,J)
0107     1033   FORMAT(I5,3F12.0)
0108     205    CONTINUE
0109            CLOSE(UNIT=3)
0110            WRITE(6,104)
0111            READ(6,101) IPR
0112            IF(IPR .EQ. 0) GO TO 211
0113            CALL DPR(NEF,0.,N,100.)
0114            XCL =XLLC
0115            YCL =YLLC
0116            DXL =DELX
0117            LNLG =NOSY
0118            ICLG =NOSX
0119            NAQLG=NAQS
0120            IWDL=NOSX
0121            IF(IROT.EQ.1) IWDL=NOSY
0122            IWDL0=NOSX
0123            IWDL1=NOSY
0124            CALL DPR(NES,0.,N,100.)
0125            XCS =XLLC
0126            YCS =YLLC
0127            DXS =DELX
0128            LNSM =NOSY
0129            ICSM =NOSX
0130            NAQSM=NAQS
0131            IWDS=NOSX
0132            IF(IROT.EQ.1) IWDS=NOSY
0133            IWDS0=NOSX
0134            IWDS1=NOSY
0135            WRITE(5,105)
0136            DO 210 J =1,NWELLS
0137            X =WELQS(1,J)
0138            Y =WELQS(2,J)
0139            QGPM =WELQS(3,J)
0140            QCFD =QGPM*192.5
```

```
0141              IL0 =(X -XCL)/DXL +1.5
0142              JL0 =(Y -YCL)/DXL +1.5+(IWLAQ(J)-1)*IWDL0
0143              IL1 =LNLG -JL0 +1
0144              JL1 =IL0+(IWLAQ(J)-1)*IWDL1
0145              IS0 =(X -XCS)/DXS +1.5
0146              JS0 =(Y -YCS)/DXS +1.5+(IWLAQ(J)-1)*IWDS0
0147              IS1 =LNSM -JS0 +1
0148              JS1 =IS0+(IWLAQ(J)-1)*IWDS1
0149              WRITE(5,106)J,X,Y,QGPM,QCFD,JL0,IL0,JL1,IL1,JS0,IS0,JS1,IS1
                  1,IWLAQ(J)
0150      210     CONTINUE
0151              CLOSE(UNIT=5)
0152              WRITE(5,107)
          C
          C                 SET WELL QS FROM WELLS.FIL
          C
0153      211     WRITE(6,110)
0154              READ(6,101) IACT
0155              IF(IACT .EQ. 0) STOP
0156              WRITE(6,109)
0157              READ(6,101) IRG
0158              NFL =NEF
0159              IF(IRG .EQ. 1) NFL =NES
0160              CALL DPR(NFL,0.,N,100.)
0161              IWD=NOSX
0162              IF(IROT.EQ.1) IWD=NOSY
0163              DO 220 IQ =1,NWELLS
0164              X =WELQS(1,IQ)
0165              Y =WELQS(2,IQ)
0166              QW =WELQS(3,IQ)*192.5
0167              ICO =(X -XLLC)/DELX +1.5+(IWLAQ(IQ)-1)*IWD
0168              JLN =(Y -YLLC)/DELX +1.5
0169              IF(IROT .EQ. 0) GO TO 215
0170              ICO =NOSY -JLN +1+(IWLAQ(IQ)-1)*IWD
0171              JLN =(X -XLLC)/DELX +1.5
0172      215     IF(ICO.LE.(IWLAQ(IQ)-1)*IWD.OR.ICO.GT.IWD+(IWLAQ(IQ)-1)*IWD)
                  1 GO TO 220
0173              IF(JLN .LE. 0 .OR. JLN .GT. LINES) GO TO 220
0174              RB =(JLN-1)*12. +4.
0175              CALL DPR(NFL,RB,H0,256.)
0176              IF(IACT.EQ.1) H0(ICO) =QW +H0(ICO)
0177              IF(IACT .EQ. 2) H0(ICO) =0.
0178              IF(IACT.EQ.3) H0(ICO)=H0(ICO)-QW
0179              CALL DPW(NFL,RB,H0,256.)
0180              CALL DPR(NFL,RB+8.,H0,256.)
0181              IF(IACT.EQ.1) H0(ICO) =QW+ H0(ICO)
0182              IF(IACT .EQ. 2) H0(ICO) =0.
0183              IF(IACT.EQ.3) H0(ICO)=H0(ICO)-QW
0184              CALL DPW(NFL,RB+8.,H0,256.)
0185      220     CONTINUE
0186              GO TO 1
          C
          C                 SET LEAKAGE COFF.
          C
0187      300     WRITE(6,112)
0188              READ(6,101) IWR
```

```
0189                NFL =NEF
0190                IF(IRR .EQ. 1) NFL =NES
0191                CALL DPR(NFL,0.,N,102,)
0192                WRITE(6,113)
0193                READ(6,103) LEAK
0194                CALL DPW(NFL,0.,N,102,)
0195                STOP
          C
0196      100       FORMAT(1X,'TRANSFER SMALL TO LARGE REGION ? :1=Y,CR=NO')
0197      101       FORMAT(I6)
0198      102       FORMAT(1X,'TRANSFER POTENTIAL TO SMALL REGION ? :1=YES,CR=NO')
0199      103       FORMAT(3F12.0)
0200      104       FORMAT(1X,'PRINT OUT WELLS AND LOCATIONS ?: 1=Y')
0201      105       FORMAT(T58,'LARGE REGION',T90,'SMALL REGION'//
                    1T55,'R=0',T71,'R=1',T87,'R=0',T103,'R=1'/
                    2' WELL',T10,'X(FT)',T20,'Y(FT)',T30,'Q(GPM)',T40,'Q(CFD)'
                    3,T50,'LINE',T58,'COLL',T66,'LINE',T74,'COLL',T82,'LINE'
                    4,T90,'COLL',T98,'LINE',T106,'COLL',T120,'AQUIFER'/)
0202      106       FORMAT(I4,4F10.0,8I8,T120,I7/)
0203      107       FORMAT(1H1)
0204      108       FORMAT(1X,'WELL QS IO ? :1=Y')
0205      109       FORMAT(1X,'LARGE OR SMALL REG. ?:CR=LG,1=SM')
0206      110       FORMAT(1X,'CR= STOP ,1= SET QS ,2= CLEAR QS '
                    1,',3= REMOVE QS :I2')
0207      111       FORMAT(1X,'CHANGE LEAKAGE COFF. ? CR=NO,1=YES')
0208      112       FORMAT(1X,'LARGE RG. =CR ,SMALL RG. =1')
0209      113       FORMAT(1X,'NEW LEAKAGE VALUE = :F6.0')
0210                END
```

```
0001            SUBROUTINE FINTP(FLDCR,FSQR,MAG,IARFG,NTP,IXCR)
0002            DIMENSION FLDCR(4),FSQR(9,9),AREA(4)
0003            INTEGER NTP(2,128,9),NTPCR(9,9),NCR(4)
0004            DATA AREA/1.,.5,.375,.625/
      C
0005            RMAG =MAG
0006            DO 5 J =1,MAG+1
0007            DO 5 I =1,MAG+1
0008            IX =IXCR +I
0009            NTPCR(I,J) =NTP(2,IX,J)
0010            FSQR(I,J) =0.
0011      5     CONTINUE
0012            NCR(1) =NTPCR(1,1)
0013            NCR(2) =NTPCR(MAG+1,1)
0014            NCR(3) =NTPCR(MAG+1,MAG+1)
0015            NCR(4) =NTPCR(1,MAG+1)
0016            NGD =0
0017            DO 10 I =1,4
0018            IF(NCR(I) .NE. 1) NGD =NGD +1
0019     10     CONTINUE
0020            IF(NGD .EQ. 0) RETURN
0021            GO TO(30,30,15,30),NGD
0022     15     DO 20 ICR =1,4
0023            IF(NCR(ICR) .EQ. 1) GO TO 25
0024     20     CONTINUE
0025            STOP
0026     25     ICA =ICR+1
0027            IF(ICA .EQ. 5) ICA =1
0028            ICB =ICR -1
0029            IF(ICB .EQ. 0) ICB =4
0030            ICC =ICR +2
0031            IF(ICC .GT. 4) ICC =ICC-4
0032            FLDCR(ICR) =FLDCR(ICA) +FLDCR(ICB) -FLDCR(ICC)
0033     30     A =FLDCR(2) -FLDCR(1)
0034            B =FLDCR(4) -FLDCR(1)
0035            C =FLDCR(3) +FLDCR(1) -FLDCR(4) -FLDCR(2)
0036            DO 35 J =1,MAG+1
0037            DO 35 I =1,MAG+1
0038            X =(I-1)/RMAG
0039            Y =(J-1)/RMAG
0040            F =FLDCR(1) +A*X +B*Y +X*Y*C
0041            IF(NTPCR(I,J) .EQ. 1) F =0.
0042            IF(IARFG .GT. 0) F =F/(RMAG*RMAG)
0043            FSQR(I,J) =F
0044     35     CONTINUE
0045            RETURN
0046            END
```

```
DB0:LISTP,LP/SH=LISTP
[1,1]BNWLIB/LB
/
ASG=TI:6,DB0:2,LP:5
ASG=DP0:4
/
```

```
        C ***** LISTP ********
        C LISTP=LISTP
        C ICHR
        C [1,1]BNWLIB,OLB/LB
        C /
        C ASG=TI:6,DB0:2,LP:5
        C ASG=DP0:4
        C /
        C PROGRAM FOR PROCESSING LIST INFORMATION FOR THE MULTI-REGION MODEL
0001            INTEGER RTIMES
        C       COMMON /LISTS/NOLRG,ISAVE(2,128),SPACE,NOHLD,IUNSAV(2,256),
        C       1SVAL(128),UNSVAL(256)
0002            COMMON /LISTS/NOHLD,IUNSAV(2,512),IEXTRA,UNSVAL(512)
0003            COMMON /RESULT/NUMSAV,RTIMES(3,100),RMIN(100),NTPLSV,ISAVNO(100)
0004            COMMON /BK0/NUNK,IHBW,NLINES,NCOLS
0005            COMMON /LIST/NLGPR,MAG,LIST(2046)
0006            COMMON /LIST1/NPAIR,MAGS,LIST1(2046)
0007            COMMON /LARGE/NLRG,LSTLRG(2,256),IG1
0008            COMMON /SMALL/NSML,LSTSML(2,512),IG2
0009            DIMENSION LSTTMP(2,11,50),VALLRG(256),VALSML(512),FNAME(7)
                1,H0(256),FA(4608),ISAVE(2,256),SVAL(256)
0010            EQUIVALENCE (FA(1),LSTTMP(1,1,1)),(FA(551),VALLRG(1))
                1,(FA(1064),VALSML(1)),(FA(2100),H0(1))
                1,(ISAVE(1,1),IUNSAV(1,1)),(SVAL(1),UNSVAL(1)),(NOHLD,NOLRGG)
0011            DATA RNONE/4HNONE/
0012            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0013            READ(6,*) IUN
0014            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0015            WRITE(6,1333)
0016    1333    FORMAT(' ENTER LIST FILE NAME OR NONE')
0017            READ(6,1334) FNAME
0018            CALL OPFILE(2,'VTTLIST',624.,LST)
0019            CALL OPFILE(2,'VTTRESULT',40006.,IANS)
0020            IRSLTO=0
0021            CALL DPR(IANS,0.,+IRSLTO,NUMSAV,602.)
0022    1334    FORMAT(7A4)
0023            IF(FNAME(1).EQ.RNONE) GO TO 1335
0024            N=ICHR(FNAME)
0025            OPEN(UNIT=4,NAME=FNAME,TYPE='OLD',READONLY)
0026            MM=2
0027            READ(4,1336) NLGPR,MAG
0028    1336    FORMAT(<MM>I3)
0029            M=((MAG+1)*2+2*2)*NLGPR+NLGPR
0030            MM=(MAG+1)*2+2*2+1
0031            READ(4,1336) (LIST(I),I=1,M)
0032            CLOSE(UNIT=4)
0033            CALL DPW(LST,0.,NLGPR,2048.)
0034    1335    CALL DPR(LST,0.,NLGPR,2048.)
0035            WRITE(6,1)
0036    1       FORMAT(' LIST PROCESSING PROGRAM'/
                1' 0 FOR SET UP LISTS'/
                1' 1 FOR INTERPLOATING TO SMALL GRID FROM LIST'/
                1' 2 FOR INTERPOLATING TO SMALL GRID FROM TIME STEPS')
0037            READ(6,2) IWCH
0038    2       FORMAT(I5)
        C PREPARE LISTS FOR SAVING AND INTERPOLATING
```

```
0039            IOFSET=(MAG+1)*2+4+1
0040            ISM=1
0041            ILG=1
0042            NLRG=NLGPR*2
0043            NSML=NLGPR*(MAG+1)*2
0044            ISTPX0=NLRG*2+NSML+NLGPR
0045            NPAIR=NLGPR
0046            MAGS=MAG
0047            IOFST1=(MAG+1)+2
0048            ILST0=0
0049            ILST1=0
0050            LSTLG=0
0051            LSTSM=0
0052     8      IF(ILST0.GE.ISTPX0) GO TO 50
0053            ITMP=1
0054            LINE0=LIST(ILST0+2)
         D      WRITE(5,550) IOFSET,ISM,ILG,NLRG,NSML,ISTPX0,
         D      1NPAIR,MAGS,IOFST1,ILST0,ILST1,LSTLG,LSTSM,ITMP,LINE0
         D550   FORMAT(5I10)
0055     9      LSTTMP(1,1,ITMP)=LIST(ILST0+1)
0056            LSTTMP(2,1,ITMP)=LIST(ILST0+2)
0057            LSTTMP(1,2,ITMP)=LIST(ILST0+3)
0058            LSTTMP(2,2,ITMP)=LIST(ILST0+4)
0059            DO 10 I=1,MAG+1
0060            LSTTMP(1,2+I,ITMP)=LIST(ILST0+5+(I-1)*2+1)
0061            LSTTMP(2,2+I,ITMP)=LIST(ILST0+5+(I-1)*2+2)
0062     10     CONTINUE
0063            ILST0=ILST0+IOFSET
0064            IF(ILST0.GE.ISTPX0) GO TO 11
0065            IF(LIST(ILST0+2).NE.LINE0) GO TO 11
0066            ITMP=ITMP+1
0067            GO TO 9
         C HAVE ITMP VALUES IN LSTTMP FOR LINE0 AND LINE0+1
0068     11     K=ILST1
         D      WRITE(5,551) (((LSTTMP(L,M,N),L=1,2),M=1,2+MAG+1),N=1,ITMP)
         D551   FORMAT(12I10)
         D      WRITE(5,550) IOFSET,ISM,ILG,NLRG,NSML,ISTPX0,
         D      1NPAIR,MAGS,IOFST1,ILST0,ILST1,LSTLG,LSTSM,ITMP,LINE0
0069            DO 20 J=1,ITMP
0070            LSTLRG(1,LSTLG+1)=LSTTMP(1,1,J)
0071            LSTLRG(2,LSTLG+1)=LSTTMP(2,1,J)
0072            LSTLG=LSTLG+1
0073            LIST1(ILST1*IOFST1+1)=LSTLG
0074            IF(LSTTMP(2,2,J).NE.LINE0) GO TO 19
0075            LSTLRG(1,LSTLG+1)=LSTTMP(1,2,J)
0076            LSTLRG(2,LSTLG+1)=LSTTMP(2,2,J)
0077            LSTLG=LSTLG+1
0078            LIST1(ILST1*IOFST1+2)=LSTLG
0079            DO 15 I=1,MAG+1
0080            LSTSML(1,LSTSM+1)=LSTTMP(1,I+2,J)
0081            LSTSML(2,LSTSM+1)=LSTTMP(2,I+2,J)
0082            LSTSM=LSTSM+1
0083            IDORK=LSTSM
0084            IF(LSTSML(1,LSTSM).EQ.0) IDORK=-1
0085            LIST1(ILST1*IOFST1+2+I)=IDORK
0086     15     CONTINUE
```

```
0087    19      ILST1=ILST1+1
0088    20      CONTINUE
0089            DO 30 L=1,MAG+1
0090            ILST1=K
0091            DO 30 J=1,ITMP
0092            IF(LSTTMP(2,2,J).EQ.LINE0) GO TO 29
0093            IF(L.GT.1) GO TO 24
0094            LSTLRG(1,LSTLG+1)=LSTTMP(1,2,J)
0095            LSTLRG(2,LSTLG+1)=LSTTMP(2,2,J)
0096            LSTLG=LSTLG+1
0097            LIST1(ILST1*IOFST1+2)=LSTLG
0098    24      CONTINUE
0099            LSTSML(1,LSTSM+1)=LSTTMP(1,L+2,J)
0100            LSTSML(2,LSTSM+1)=LSTTMP(2,L+2,J)
0101            LSTSM=LSTSM+1
0102            IDORK=LSTSM
0103            IF(LSTSML(1,LSTSM).EQ.0) IDORK=-1
0104            LIST1(ILST1*IOFST1+2+L)=IDORK
0105    29      ILST1=ILST1+1
0106    30      CONTINUE
        D       WRITE(5,552)
        D552    FORMAT(' AFTER 30 CONTINUE')
        D       WRITE(5,551) ((LSTLRG(M,N),M=1,2),N=1,LSTLG)
        D       WRITE(5,551) ((LSTSML(M,N),M=1,2),N=1,LSTSM)
0107            GO TO 8
0108    50      CONTINUE
0109            CALL DPW(LST,8.,NLRG,514.)
0110            IF(IWCH.NE.0) CALL DPW(LST,11.,NSML,1026.)
0111            CALL DPW(LST,616.,NPAIR,2048.)
        D       WRITE(5,2999)
        D       WRITE(5,3001)NLRG
        D3001   FORMAT(2I5)
        D       WRITE(5,3002) ((LSTLRG(I,J),I=1,2),J=1,NLRG)
        D       WRITE(5,2999)
        D3002   FORMAT(5(2I6,10X))
        D       WRITE(5,3001) NSML
        D       WRITE(5,3002) ((LSTSML(I,J),I=1,2),J=1,NSML)
        D       WRITE(5,2999)
        D       WRITE(5,3001) NPAIR,MAGS
        D       WRITE(5,3003) LIST1
        D3003   FORMAT(20I6)
0112            IF(IWCH.EQ.1) GO TO 1000
0113            IF(IWCH.EQ.2)GO TO 2000
0114            STOP
        C DO THE INTERPOLATION
0115    1000    CONTINUE
0116            CALL DPR(LST,616.,NPAIR,2048.)
0117            IOFST1=(MAGS+1)+2
        D       WRITE(5,2998) NPAIR,MAGS
        D2998   FORMAT(10(2I4,2X))
        D       WRITE(5,2998) LIST1
0118            DO 1020 J=1,NUMSAV
0119            CALL DPR(LST,(J-1)*2.+16.,VALLRG,512.)
        D       WRITE(5,2999)
        D2999   FORMAT(1H1)
        D       WRITE(5,3000) VALLRG
```

```
        D       WRITE(5,2999)
0120            DO 1010 I=1,NPAIR
0121            Y2=VALLRG(LIST1((I-1)*IOFST1+2))
0122            Y1=VALLRG(LIST1((I-1)*IOFST1+1))
0123            DO 1005 K=1,MAGS+1
0124            I1=LIST1((I-1)*IOFST1+2+K)
0125            IF(I1.EQ.-1) GO TO 1005
0126            VALSML(I1)=(Y2-Y1)*FLOAT(K-1)/FLOAT(MAGS)+Y1
0127    1005    CONTINUE
0128    1010    CONTINUE
0129            CALL DPW(LST,(J-1)*4.+216.,VALSML,1024.)
        D       WRITE(5,3000) VALSML
        D3000   FORMAT(10F10.2)
0130    1020    CONTINUE
0131            STOP
0132    2000    CONTINUE
0133            SAVADR=16.
0134            NOLRGG=0
0135            CALL DPR(LST,8.,NOLRGG,514.)
        D       WRITE(5,*) 'NOLRGG,ISAVE(2,256) ARRAY',NOLRGG
        D       WRITE(5,1257) ISAVE
        D1257   FORMAT(1X,40I3)
0136            CALL DPFILE(2,'VTTFIELDS',2401.,IPTR)
0137            CALL DPR(IPTR,0.,NUNK,4.)
        C CODING REQUIRED FOR COMPRESSION OF TIME STEP AND TIME PLANE
        C POTENTIAL DATA IN THE RESULT FILE AND THE CODING TO RETRIVE
        C THIS COMPRESSED DATA AND PUT INTO THE VTTFIELDS FILE
        C
        C THIS FOLLOWING PROCEEDURE NEEDS ONLY BE DONE ONCE FOR AND GIVEN
        C SYSTEM IT CALCULATES SPACE REQUIRED AND THE NUMBER OF PAGES OF
        C DATA NEEDED FOR THE COMPRESSION
0138            NLIN=(4608-128)/NCOLS
0139            IF(NLIN.GT.NLINES) NLIN=NLINES
0140            NPGS=NLINES/NLIN
0141            IF(MOD(NLINES,NLIN).NE.0) NPGS=NPGS+1
        C NPG=TOTAL NUMBER OF PAGES REQUIRED
        C NLIN=NUMBER OF MATRIX LINES PER PAGE NOT COUNTING THE 1 LINE OVERLAP
        C NCOLS=TOTAL NUMBER OF COLUMNS PER MATRIX LINE
        C NLINES=TOTAL NUMBER OF MATRIX LINES
        C TOTAL SIZE OF BUFFER REQUIRED FOR THIS PROCEEDURE IS 4608 REAL WORDS
        C TO ALLOW FOR THE ONE LINE OVERLAP
        C NOW CALCULATE THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR STORAGE
        C OF A TIME PLANE
        C
        C ADJUST THE LINE COUNT FOR OVERLAP
0142            IF(NLIN.NE.NLINES) NLINA=NLIN+1
        C CALCULATE THE DISK BLOCKS REQUIRED FOR ONE FULL PAGE
0143            IBLKPP=NLINA*NCOLS/128
0144            IF(MOD(NLINA*NCOLS,128).NE.0) IBLKPP=IBLKPP+1
0145            ITOTBK=IBLKPP*(NPGS-1)
        C ITOTBK IS THE BLOCKS REQUIRED FOR FULL PAGES NOW CALCULATE
        C THE BLOCKS REQUIRED FOR THE LAST PAGE
0146            LINLFT=NLINES-NLIN*(NPGS-1)
0147            IBLKLF=LINLFT*NCOLS/128
0148            IF(MOD(LINLFT*NCOLS,128).NE.0) IBLKLF=IBLKLF+1
0149            ITOTBK=ITOTBK+IBLKLF
```

```
        C NOW ITOTBK IS THE TOTAL NUMBER OF DISK BLOCKS REQUIRED FOR TIME PLANE
        C SAVING.
        C
0150            RNWDS=NCOLS*2.
0151            DO 1393 IPLN=1,NUMSAV
0152            NOD=1
0153            DO 1388 IKK=1,NOLRGG
0154            SVAL(IKK)=0.
0155    1388    CONTINUE
0156            IF(NOD.GT.NOLRGG) GO TO 1390
0157            DO 1392 IPG=1,NPGS
0158            ISTLN=(IPG-1)*NLIN+1
0159            ISTPLN=ISTLN+NLIN-1
0160            ISTPLN=MIN(ISTPLN,NLINES)
0161            IF(IPG.NE.NPGS) ISTPLN=ISTPLN+1
0162            BUFWDS=(ISTPLN-ISTLN+1)*NCOLS*2
0163            ADR=3.+ITOTBK*(IPLN-1)+(IPG-1)*IBLKPP
0164            CALL DPR(IANS,ADR,FA,BUFWDS)
0165    1391    IF(ISAVE(2,NOD).GT.ISTPLN) GO TO 1392
0166            IF(ISAVE(2,NOD).LT.ISTLN) WRITE(6,*) 'ERROR AFTER 1391'
                1,ISAVE(2,NOD),ISTLN,ISTPLN
0167            IF(ISAVE(2,NOD).LT.ISTLN) STOP
0168            IPOS=(ISAVE(2,NOD)-ISTLN)*NCOLS+ISAVE(1,NOD)
0169            SVAL(NOD)=FA(IPOS)
0170            NOD=NOD+1
0171            IF(NOD.GT.NOLRGG) GO TO 1390
0172            GO TO 1391
0173    1392    CONTINUE
0174    1390    CALL DPW(LST,SAVADR,SVAL,512.)
        D       IF(IPLN.NE.1.OR.IPLN.NE.NUMSAV) GO TO 8967
        D       WRITE(5,*) IPLN
        D       WRITE(5,8966) (SVAL(IU),IU=1,NOLRGG)
        D8966   FORMAT(10F13.2)
        D8967   CONTINUE
0175    1389    CONTINUE
0176            SAVADR=SAVADR+2.
0177    1393    CONTINUE
0178            GO TO 1000
0179            END
```

```
DP0:ACHEK=ACHEK
[1,1]6NWLI8/LB
/
ASG=TI:6,LP:5,DB0:2
/
```

```
        C       *****ACHEK*****
        C       PROGRAM: ACHEK
        C       FUNCTION: PRINTS OUT HEADER BLOCK,ARRAY LINES,OR B.C. LIST
        C       AUTHOR: H. P. FOOTE
        C
0001            BYTE STAT,DOIT(2),TITLE
0002            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,CONV,BIAS,NAQS,IOXFER
                1,IROT,MAXIT,DELX,IAUTO,OMEGA,NTMPLN,NOTIMS,ITROS,IWD,TIME,STIME
                2,ETIME,IYEAR,MONTH,IDAY,IHOUR,IMIN,IYEARL,MONTHL,IDAYL
                3,IHOURL,MINL,NOSX,NOSY,NAQFRS,NUCONF,DOIT,STOCF,TITLE(80)
                4,LEAK,SEALVL,OFSET,ITVN,ILGSML,ISMLSM,XLLC,YLLC,ISBLVL,MAG,ISPC
0003            DIMENSION BUFF(128,12),H(128),BTM(128),K(128),Q(128)
0004            DIMENSION TOP(128),TRN(128),OCN(128),OH(128),ST(128),Q2(128)
0005            REAL K,LEAK
0006            INTEGER NTP(256),NNO(256),IRB(12),ITYP(12),ICLNO(128),LIST(2048)
0007            EQUIVALENCE (BUFF(1,1),H),(BUFF(1,2),BTM),(BUFF(1,3)
                1,K),(BUFF(1,4),Q),(BUFF(1,5),NTP),(BUFF(1,6),NNO)
                2,(BUFF(1,7),TOP),(BUFF(1,8),TRN),(BUFF(1,9),OCN)
                3,(BUFF(1,10),OH),(BUFF(1,11),ST),(BUFF(1,12),Q2)
                4,(BUFF(1,1),LIST)
0008            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0009            READ(6,*) IUN
0010            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0011            WRITE(6,121)
0012            READ(6,102) IFLD
0013            IF(IFLD.EQ.0) CALL OPFILE(2,'VTTFIELDS',2401.,NEF)
0014            IF(IFLD.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,NEF)
0015            CALL OPR(NEF,0.,N,102.)
0016            WRITE(6,120)
0017            READ(6,102) IRB
0018            DO 402 I =1,2
0019    402     ITYP(I) =0
0020            DO 405 IP =1,2
0021            IND =IRB(IP)
0022            IF(IND .LT. 1 .OR. IND .GT. 2) GO TO 405
0023            ITYP(IND) =1
0024    405     CONTINUE
0025            IF(ITYP(1) .EQ. 1) GO TO 200
0026    406     IF(ITYP(2) .EQ. 1) GO TO 250
0027    1       WRITE(6,100) N,M,LINES,ICOL,NIT,ITCT
0028    3       WRITE(6,101)
0029            READ(6,102,END=1300) IBG,IEND
0030            WRITE(6,103)
0031            WRITE(6,113)
0032            READ(6,102) IRB
0033            DO 2 I =1,12
0034    2       ITYP(I) =0
0035            DO 5 IP =1,12
0036            IND =IRB(IP)
0037            IF(IND .LT. 1 .OR. IND .GT. 12) GO TO 5
0038            ITYP(IND) =1
0039    5       CONTINUE
0040            NOVR =ICOL/12
0041            IF(MOD(ICOL,12) .NE. 0) NOVR =NOVR +1
0042            DO 10 I =1,128
0043    10      ICLNO(I) =I
```

```
0044            WRITE(5,100) N,M,LINES,ICOL,NIT,ITCT
0045            DO 20 ILNS =IEND,IBG,-1
0046            RB =(ILNS-1)*12 +1
0047            WRITE(5,112) ILNS
0048            CALL DPR(NEF,RB,BUFF,12,*256,)
0049            DO 20 IG =1,NOVR
0050            IA =(IG-1)*12 +1
0051            IB =IA +11
0052            I1 =2*IA-1
0053            I2 =2*IB-1
0054            I3 =I1+1
0055            I4 =I2+1
0056            WRITE(5,111)(ICLNO(I),I =IA,IB)
0057            IF(ITYP(1) .EQ. 1) WRITE(5,104)(H(I),I =IA,IB)
0058            IF(ITYP(2) .EQ. 1) WRITE(5,105)(BTM(I),I =IA,IB)
0059            IF(ITYP(3) .EQ. 1) WRITE(5,106)(K(I),I =IA,IB)
0060            IF(ITYP(4) .EQ. 1) WRITE(5,107)(Q(I),I =IA,IB)
0061            IF(ITYP(5) .EQ. 1) WRITE(5,108)(NTP(I),I =I1,I2,2)
0062            IF(ITYP(5) .EQ. 1) WRITE(5,109)(NTP(I),I =I3,I4,2)
0063            IF(ITYP(6) .EQ. 1) WRITE(5,110)(NNO(I),I =IA,IB)
0064            IF(ITYP(7) .EQ. 1) WRITE(5,114)(TOP(I),I =IA,IB)
0065            IF(ITYP(8) .EQ. 1) WRITE(5,115)(TRN(I),I =IA,IB)
0066            IF(ITYP(9) .EQ. 1) WRITE(5,116)(OCN(I),I =IA,IB)
0067            IF(ITYP(10) .EQ. 1) WRITE(5,117)(OH(I),I =IA,IB)
0068            IF(ITYP(11) .EQ. 1) WRITE(5,118)(ST(I),I =IA,IB)
0069            IF(ITYP(12) .EQ. 1) WRITE(5,119)(Q2(I),I =IA,IB)
0070     20     CONTINUE
0071            CLOSE(UNIT=5)
0072            IF(M.NE.0) GO TO 1
0073     1300   STOP
0074     100    FORMAT(8X,'N',9X,'M',9X,'LINES',5X,'ICOL',6X,'NIT',7X,'ITCT'/
                1,1X,6I10/)
0075     112    FORMAT(/10X,'LINE NO. =',I4/)
0076     101    FORMAT(1X,'START LINE ,STOP LINE :2I4')
0077     102    FORMAT(12I4)
0078     103    FORMAT(1X,'WHICH ARRAYS ?  H    BTM  K  Q  NTP  NNO  TOP  TRN
                1  OCN  OPT  STO  Q2')
0079     113    FORMAT(1X'                 1     2   3  4   5    6    7    8
                1    9   10   11   12')
0080     120    FORMAT(1X,'HEADER OUTPUT ,PRINT B, C, LIST'/
                1,'      1                  2')
0081     121    FORMAT(1X,'LARGE OR SMALL FIELD ? ICR=LG,1=SM')
0082     122    FORMAT(1X,'BOUNDRY NODE LIST'/1X,'TOTAL PAIRS=',I4
                1/1X,'MAGNIFACTION=',I4/)
0083     123    FORMAT(/1X,'PAIR',I4,' X0',I4,' Y0',I4,' X1',I4,' Y1',I4)
0084     124    FORMAT(1X,'NO. OF SMALL PAIRS',I4)
0085     125    FORMAT(1X,9('[',I2,',',I2,']'))
         C
         C              HEADER OUTPUT
         C
0086     200    WRITE(5,300) TITLE
0087            WRITE(5,301) N
0088            WRITE(5,302) M
0089            WRITE(5,303) LINES
0090            WRITE(5,304) ICOL
0091            WRITE(5,305) NIT
```

```
0092            WRITE(5,306) ITCT
0093            WRITE(5,307) CONV
0094            WRITE(5,308) BIAS
0095            WRITE(5,309) NAGS
0096            WRITE(5,310) IOXFER
0097            WRITE(5,311) IROT
0098            WRITE(5,312) MAXIT
0099            WRITE(5,313) DELX
0100            WRITE(5,314) IAUTO
0101            WRITE(5,315) OMEGA
0102            WRITE(5,316) NTMPLN
0103            WRITE(5,317) NOTIMS
0104            WRITE(5,318) ITROS
0105            WRITE(5,319) IWO
0106            WRITE(5,320) TIME
0107            WRITE(5,321) STIME
0108            WRITE(5,322) ETIME
0109            WRITE(5,323) IYEAR,MONTH,IDAY,IHOUR,IMIN
0110            WRITE(5,324) IYEARL,MONTHL,IDAYL,IHOURL,MINL
0111            WRITE(5,325) NOSX
0112            WRITE(5,326) NOSY
0113            WRITE(5,327) NAQFRS
0114            WRITE(5,328) NUCONF
0115            WRITE(5,329) DOIT(1),DOIT(2)
0116            WRITE(5,330) STOCF
0117            WRITE(5,331) LEAK
0118            WRITE(5,332) SEALVL
0119            WRITE(5,333) OFSET
0120            WRITE(5,334) ITVN
0121            WRITE(5,335) ILGSML
0122            WRITE(5,336) ISMLSM
0123            WRITE(5,337) XLLC
0124            WRITE(5,338) YLLC
0125            WRITE(5,339) ISBLVL
0126            WRITE(5,340) MAG
0127            WRITE(5,350)
0128            CLOSE(UNIT=5)
0129            GO TO 406
        C
        C               8. C. LISTING
        C
0130    250     CALL DPFILE(2,'VTTLIST',624.,NEL)
0131            CALL DPR(NEL,0.,LIST,2048.)
0132            NPAR =LIST(1)
0133            MAG =LIST(2)
0134            WRITE(5,122)NPAR,MAG
0135            DO 255 IL =1,NPAR
0136            INOX =(IL-1)*(2*(MAG+1)+5)+3
0137            WRITE(5,123) IL,LIST(INOX),LIST(INOX+1),LIST(INOX+2)
                1,LIST(INOX+3)
0138            WRITE(5,124) LIST(INOX+4)
0139            IA =INOX+5
0140            IB =IA+ 2*(MAG+1) -1
0141            WRITE(5,125)(LIST(I),I =IA,IB)
0142    255     CONTINUE
0143            CLOSE(UNIT=5)
```

```
0144              GO TO 1
0145     104      FORMAT(1X,'POTENTIAL =',12F10.3)
0146     105      FORMAT(1X,'BOTTOM    =',12F10.3)
0147     106      FORMAT(1X,'HYD. COND.=',12F10.3)
0148     107      FORMAT(1X,'Q-SOURCE  =',12F10.0)
0149     108      FORMAT(1X,'NODE #S   =',I7,11I10)
0150     109      FORMAT(1X,'NODE TYPES=',O7,11O10)
0151     110      FORMAT(1X,'NODE LOC  =',I7,11I10)
0152     111      FORMAT(/1X,'COLL. POS.=',I7,11I10)
0153     114      FORMAT(1X,'TOP OF AQF=',12F10.3)
0154     115      FORMAT(1X,'TRAN COFF.=',12F10.3)
0155     116      FORMAT(1X,'OCEAN TRNS=',12F10.3)
0156     117      FORMAT(1X,'OLD PONTAL=',12F10.3)
0157     118      FORMAT(1X,'STORAGE   =',12F10.3)
0158     119      FORMAT(1X,'Q2 FIELD  =',12F10.0)
0159     300      FORMAT(90X,80A1//)
0160     301      FORMAT(1X,'NO. OF UNKNOWNS',T35,I5)
0161     302      FORMAT(1X,'HALF BANDWIDTH',T35,I5)
0162     303      FORMAT(1X,'NO. OF LINES',T35,I5)
0163     304      FORMAT(1X,'NO. OF COLUMNS',T35,I5)
0164     305      FORMAT(1X,'NO. OF ITTERATIONS',T35,I5)
0165     306      FORMAT(1X,'ITTERATION COUNT',T35,I5)
0166     307      FORMAT(1X,'CONVERGENCE LIMIT',T35,F10.4)
0167     308      FORMAT(1X,'ELEVATION BIAS',T35,F10.2)
0168     309      FORMAT(1X,'NO. OF AQUIFERS',T35,I5)
0169     310      FORMAT(1X,'OCEAN TRANS. 1=YES, 0=NO',T35,I5)
0170     311      FORMAT(1X,'ROTATION 1=YES, 0=NO',T35,I5)
0171     312      FORMAT(1X,'MAX NO. OF ITTERATIONS',T35,I5)
0172     313      FORMAT(1X,'NODE SPACING',T35,F10.2)
0173     314      FORMAT(1X,'AUTO OPERATION 1=YES ,0 =NO',T35,I5)
0174     315      FORMAT(1X,'OVER RELAXATION FACTOR',T35,F6.2)
0175     316      FORMAT(1X,'NO. OF SAVED TIME PLANES',T35,I5)
0176     317      FORMAT(1X,'NO. OF Q-TIMES ',T35,I5)
0177     318      FORMAT(1X,'TRANSIENT OR STST.',T35,I5)
0178     319      FORMAT(1X,'AQUIFER WIDTH',T35,I5)
0179     320      FORMAT(1X,'DAYS SINCE START OF TEST',T35,F10.0)
0180     321      FORMAT(1X,'STARTING TIME',T35,F10.0)
0181     322      FORMAT(1X,'ENDING TIME',T35,F10.0)
0182     323      FORMAT(1X,'STARTING DATE:/Y/M/D/H/M/',T35,'/',I4,'/',I2,'/'
                  1,I2,'/',I2,'/',I2,'/')
0183     324      FORMAT(1X,'ENDING DATE:/Y/M/D/H/M/',T35,'/',I4,'/',I2,'/'
                  1,I2,'/',I2,'/',I2,'/')
0184     325      FORMAT(1X,'X -NODES',T35,I5)
0185     326      FORMAT(1X,'Y -NODES',T35,I5)
0186     327      FORMAT(1X,'NO. OF AQUIFERS',T35,I5)
0187     328      FORMAT(1X,'NO. OF UNCONFINED NODES',T35,I5)
0188     329      FORMAT(1X,'DOIT(1) =',A1,5X,'DOIT(2) =',I5)
0189     330      FORMAT(1X,'STORAGE MULT.',T35,F10.1)
0190     331      FORMAT(1X,'LEAKAGE COFF.',T35,F10.2)
0191     332      FORMAT(1X,'MEAN SEALEVEL',T35,F10.2)
0192     333      FORMAT(1X,'LEAKAGE OFFSET',T35,F10.2)
0193     334      FORMAT(1X,'TIME VARING NODES (1=Y,0=N) ',T35,I5)
0194     335      FORMAT(1X,'FIELD :(0=LARGE ,1=SMALL) ',T35,I5)
0195     336      FORMAT(1X,'STORE B.C. FOR SMALL FIELD (1=Y,0=N)',T35,I5)
0196     337      FORMAT(1X,'X COORDINATE OF L.L.C.(FT.)',T35,F10.1)
0197     338      FORMAT(1X,'Y COORDINATE OF L.L.C.(FT.)',T35,F10.1)
0198     339      FORMAT(1X,'SUBLEVEL (0=STARTING FIELD)',T35,I5)
0199     340      FORMAT(1X,'MAGNIFICATION FACTOR',T35,I5)
0200     350      FORMAT(1H1)
0201              END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

MXPLT

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DB0:MXPLT=MXPLT
[1,1]BNWLIB/LB
/
UNITS=8
ACTFIL=5
ASG=TG:7
ASG=TI:6
ASG=LP:5
ASG=DB0:2
ASG=DP0:8
```

```
        C       ******MXPLOT*****
        C       PROGRAM: MXPLOT
        C       FUNCTION: NODE TYPE PLOTTING ,CONTOURING ,STREAM LINES
        C                 ,AND POTENTIAL PROFILES
        C       AUTHOR: M. P. FOOTE
        C
0001            COMMON /BK0/ H
0002            DIMENSION H(128,16),HCR(4),H0(128),H2(128),Q0(128),HAQ(128,8,2)
                1,STREAM(4,50)
0003            INTEGER BDTYP(27),NTP(2,128),NNO(256),NDPLT(5),IBF(5)
                1,NTP1(2,128),MAP(4),NTPCR(4),NPLNS(2),IHD(100),IROTT
                2,MAPC(3)
0004            REAL K,MXDIF,XLLC,YLLC,PRFILE(301,3),DELX,KAV
0005            BYTE STAT,IOS,NAMM(28)
0006            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,IROT
0007            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0008            COMMON /BLK1/ NODE,NEF,KAV
0009            EQUIVALENCE (H(1,5),NTP(1,1)),(H(1,6),NNO),(H(1,13),NTP1(1,1))
                1,(H(1,1),H2),(H(1,9),H0),(H(1,12),Q0),(IHD(96),XLLC)
                2,(IHD(98),YLLC),(IHD(15),DELX),(IHD(9),BIAS),(IHD(13),IROTT)
                3,(H(1,1),HAQ(1,1,1)),(IHD(11),NAQS),(IHD(40),NDSX)
                4,(IHD(41),NDSY)
0010            DATA BDTYP/286,286,826,1865,864,3467,246,1245,824,3287
                1,286,865,467,346,245,124,823,287,186,8245,7824,6823
                2,5682,4681,3468,2467,1246/
0011            DATA MAP/1,3,7,2/
0012            DATA MAPC/1,7,2/
        C
        C
0013            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DBOI'
0014            READ(6,*) IUN
0015            IF(IUN.NE.3) CALL ASNLUN(2,'DP',IUN,ISTAT)
0016            WRITE(6,116)
0017            READ(6,101) IFLD
0018            IF(IFLD.EQ.0) CALL OPFILE(2,'VTTFIELDS',2401.,NEF)
0019            IF(IFLD.EQ.1) CALL OPFILE(2,'VTTSMALL',2401.,NEF)
        C       CALL OPFILE(2,'VTTRESULT',40006.,NER)
0020            CALL OPR(NEF,0.,N,6.)
        C       CALL OPR(NER,IFLD*20003.,NPLNS,2.)
0021    1       CALL OPR(NEF,0.,IHD,100.)
0022            IRT=IROTT
0023            WRITE(6,123)
0024    123     FORMAT(' WHICH OUTPUT DEVICE?'/
                1' SCOPE=CR  XY:=1  DISK=2')
0025            READ(6,101) IDEV
0026            IF(IDEV.EQ.0) CALL ASNLUN(7,'TG',0,STAT)
0027            IF(IDEV.EQ.1) CALL ASNLUN(7,'XY',0,STAT)
0028            IF(IDEV.EQ.2) CALL ASNLUN(7,'DP',0,STAT)
0029            WRITE(6,117)
0030            READ(6,101) ICRS
0031            IF(ICRS .EQ. 1) GO TO 500
0032            PI =3.14159265
0033            WRITE(6,*) 'NODE AND/OR CONTOUR PLOTS AND STREAMLINES AND/OR
                1 TRAVEL TIMES'
0034            WRITE(6,*) 'MANUAL OPERATION =CR   AUTOMATIC =1'
```

```
0035            READ(6,101) IMNU
0036            IUT=6
0037            IF(IMNU.NE.1) GO TO 8329
0038            IUT=8
0039            WRITE(6,*) 'ENTER A NAME FOR THE AUTOMATIC CONTROL FILE'
0040            READ(6,1382) NAMM
0041     1382   FORMAT(28A1)
0042            N=ICHR(NAMM)
0043            OPEN(UNIT=8,NAME=NAMM,TYPE='OLD',
                1READONLY)
0044     8329   CONTINUE
0045            IF(IDEV.NE.2)
                1OPEN(UNIT=7,NAME='MXPLT.DAT',FORM='UNFORMATTED'
                1,TYPE='NEW')
0046            IF(IDEV.NE.2) GO TO 1386
0047            WRITE(6,*) 'ENTER THE NAME OF THE DISK PLOT FILE'
0048            READ(6,1382) NAMM
0049            N=ICHR(NAMM)
0050            OPEN(UNIT=7,NAME=NAMM,FORM='UNFORMATTED'
                1,TYPE='NEW')
0051     1386   CONTINUE
0052            IWD=NDSX
0053            NDSXX=NDSX
0054            NDSYY=NDSY
0055            IF(IRT.EQ.0) GO TO 1233
0056            IWD=NDSY
0057            NDSXX=NDSY
0058            NDSYY=NDSX
0059     1233   CONTINUE
0060            ISAQ=1
0061            IEAQ=NAQS
0062            ISCL=1
0063            IECL=NDSXX
0064            ISLNN=1
0065            IELNN=NDSYY
0066            WRITE(6,*) 'THIS IS A',NAQS,' AQUIFER SYSTEM'
0067            WRITE(6,*) 'EACH AQUIFER IS',NDSXX,' COLUMNS'
0068            WRITE(6,*) '              BY',NDSYY,' LINES'
0069            WRITE(6,*) 'DO YOU WISH TO DISPLAY EVERYTHING CR=YES, 1=NO'
0070            READ(IUT,101) IALL
0071            IF(IALL.EQ.0) GO TO 12344
0072            IF(NAQS.EQ.1) GO TO 12343
0073            WRITE(6,*) 'ENTER STARTING AND ENDING AQUIFER NUMBER'
0074            READ(IUT,*) ISAQ,IEAQ
0075     12343  CONTINUE
0076            WRITE(6,*) 'ENTER STARTING AND ENDING COLUMN NUMBERS'
0077            READ(IUT,*) ISCL,IECL
0078            WRITE(6,*) 'ENTER STARTING AND ENDING LINE NUMBERS'
0079            READ(IUT,*) ISLNN,IELNN
0080     12344  CONTINUE
0081            WRITE(6,115)
        C ENTER SCALE FACTOR
0082            READ(IUT,*) SF
0083            DH =.3
0084            WRITE(6,100)
        C INPUT ROTATION
```

```
0085            READ(IUT,101) IROT
0086            CALL PLOTS(0.,0.,7)
0087            CALL FACTOR(SF)
0088            CALL PLOT(-DH/2.,-DH/2.,-3)
0089            CALL NEWPEN(2)
0090            CALL SYMBOL(0.,0.,.1,3,0.,-1)
        C       WD =ICOL*DH
        C       HI =LINES*DH
0091            ICOLN=(IEAQ-ISAQ+1)*(IECL-ISCL+1)
0092            WD=ICOLN*DH
0093            LINESN=(IELNN-ISLNN+1)
0094            HI=LINESN*DH
0095            CALL PLOTT(0.,0.,3)
0096            CALL PLOTT(WD,0.,2)
0097            CALL PLOTT(WD,HI,2)
0098            CALL PLOTT(0.,HI,2)
0099            CALL PLOTT(0.,0.,2)
0100            ACHR =.2
0101            IF(IROT .EQ. 1) GO TO 1010
0102            YP =-.75*DH
        C       DO 1001 I =1,ICOL
        C       X =I
        C       XP =(X-.5)*DH
0103            DO 1001 IAQ=ISAQ,IEAQ
0104            XS=(IAQ-ISAQ)*(IECL-ISCL+1)
0105            DO 1001 I=ISCL,IECL
0106            X=(IAQ-1)*NDSXX+I
0107            XP=(XS+I-ISCL+1.-.6)*DH
0108    1001    CALL NUMBER(XP,YP,.1,X,0.,-1)
0109            XP =-DH
        C       DO 1002 J =1,LINES
0110            DO 1002 J=ISLNN,IELNN
0111            Y =J
        C       YP =(Y-.5)*DH
0112            YP=(J-ISLNN+1.-.6)*DH
0113    1002    CALL NUMBER(XP,YP,.1,Y,0.,-1)
0114            SZ1=WD/14.
0115            SZ1=MIN(HI/12.,SZ1)
0116            SZ1=MIN(ACHR,SZ1)
0117            XM =WD/2. -.8571*SZ1*14./2.
0118            YM =HI/2. -.8571*SZ1*12./2.
0119            CALL SYMBOL(XM,-2.*DH,SZ1,'COLUMN NUMBERS',0.,14)
0120            CALL SYMBOL(-2.*DH,YM,SZ1,'LINE NUMBERS',90.,12)
0121            GO TO 1020
0122    1010    YP =-.75*DH
        C       DO 1011 J =1,LINES
0123            DO 1011 J=ISLNN,IELNN
0124            X =J
        C       XP =(X-.5)*DH
0125            XP=(J-ISLNN+1.-.6)*DH
0126    1011    CALL NUMBER(XP,YP,.1,X,0.,-1)
0127            XP =-DH
        C       DO 1012 I =1,ICOL
        C       Y =I
        C       YN =ICOL +1 -I
        C       YP =(Y-.5)*DH
```

```
        C1012   CALL NUMBER(XP,YP,.1,YN,0.,-1)
0128            DO 1012 IAQ=ISAQ,IEAQ
0129            YS=(IAQ-ISAQ)*(IECL-ISCL+1)
0130            DO 1012 I=ISCL,IECL
0131            Y=(IEAQ-IAQ)*NDSXX+(IECL-(I-ISCL))
0132            YP=(YS+I-ISCL+1.-.6)*DH
0133    1012    CALL NUMBER(XP,YP,.1,Y,0.,-1)
0134            SZ1=WD/14.
0135            SZ1=MIN(HI/12.,SZ1)
0136            SZ1=MIN(ACHR,SZ1)
0137            XM =HI/2. -.8571*SZ1*12./2.
0138            YM =WD/2. -.8571*SZ1*14./2.
0139            CALL SYMBOL(XM,-2.*DH,SZ1,'LINE NUMBERS',0.,12)
0140            CALL SYMBOL(-2.*DH,YM,SZ1,'COLUMN NUMBERS',90.,14)
0141    1020    CONTINUE
0142            DO 2 I =1,5
0143    2       NDPLT(I) =0
0144            CALL PLTOUT
0145            WRITE(6,102)
        C DESIRE NODE TYPE PLOT
0146            READ(IUT,101) IP
0147            IF(IP .EQ. 0) GO TO 200
0148            WRITE(6,103)
        C WHAT KIND OF NODE PLOT
0149            WRITE(6,104)
0150            READ(IUT,101) IBF
0151            DO 5 I =1,5
0152            IND =IBF(I)
0153            IF(IND .LT. 1) GO TO 5
0154            IF(IND .GT. 5) GO TO 5
0155            NDPLT(IND) =1
0156    5       CONTINUE
        C
        C                       NODE TYPE PLOTS
        C
        C       DO 90 ILN =1,LINES
0157            DO 90 ILN=ISLNN,IELNN
0158            RB =(ILN-1)*12 +5
0159            CALL DPR(NEF,RB,NTP,256.)
        C       DO 90 JJ =1,ICOL
        C       J=JJ
        C       IF(MOD(ILN,2).EQ.0) J=ICOL-JJ+1
        C       X =(J-1)*DH +DH/2.
        C       Y =(ILN-1)*DH +DH/2.
0160            Y=(ILN-ISLNN+.5)*DH
0161            DO 90 IAQ=ISAQ,IEAQ
0162            XS=(IAQ-ISAQ)*(IECL-ISCL+1)
0163            DO 90 ICLP=ISCL,IECL
0164            J=(IAQ-1)*NDSXX+ICLP
0165            IF(MOD(ILN,2).EQ.0)
                1J=(IEAQ-(IAQ-ISAQ))*NDSXX-(NDSXX-(IECL-(ICLP-ISCL)))
0166            X=(XS+ICLP-ISCL+.5)*DH
0167            IF(MOD(ILN,2).EQ.0) X=((IEAQ-IAQ)*(IECL-ISCL+1)+IECL-ICLP)*DH+.5*DH
0168            NDTYP =NTP(2,J)
0169            NODE =2
0170            IF(NDTYP .EQ. 0) GO TO 40
```

```
0171              IF(NDTYP .EQ. 1) GO TO 50
0172              IF(NDTYP .EQ.  2) GO TO 60
0173              IF(NDPLT(1) .EQ. 0) GO TO 30
0174              INDX =NDTYP-4
0175              IKND =BDTYP(INDX)
0176              NODE =1
0177              CALL PLTIT(X,Y,0,IKND)
0178     30       IF(NDPLT(2) .EQ. 0) GO TO 80
0179              INDX =NDTYP-4
0180              IKND =BDTYP(INDX)
0181              NODE =2
0182              CALL PLTIT(X,Y,0,IKND)
0183              GO TO 80
0184     40       IF(NDPLT(4) .EQ. 0) GO TO 80
0185              IKND =BDTYP(1)
0186              CALL PLTIT(X,Y,4,IKND)
0187              GO TO 80
0188     50       IF(NDPLT(1) .EQ. 0) GO TO 80
0189              XW =DH/10.
0190              CALL PLOTT(X-XW,Y,3)
0191              CALL PLOTT(X+XW,Y,2)
0192              CALL PLOTT(X,Y-XW,3)
0193              CALL PLOTT(X,Y+XW,2)
0194              GO TO 90
0195     60       IF(NDPLT(3) .EQ. 0) GO TO 80
0196              IKND =BDTYP(2)
0197              CALL PLTIT(X,Y,4,IKND)
0198              IKND =BDTYP(3)
0199              CALL PLTIT(X,Y,4,IKND)
0200              GO TO 90
0201     80       IF(IROT .NE. 0) GO TO 90
0202              IF(NDPLT(5) .EQ. 0) GO TO 90
0203              IF(NDTYP .EQ. 1) GO TO 90
0204              RN =NTP(1,J)
0205              XN = X -.42*DH
0206              CALL NUMBER(XN,Y,.06,RN,0.,-1)
0207     90       CONTINUE
0208              CALL PLTOUT
         C
         C                        CONTOUR PLOTTING
         C
0209     200      WRITE(6,105)
         C DO YOU DESIRE CONTOUR PLOT
0210              READ(IUT,101) IC
0211              IF(IC .EQ. 0) GO TO 3000
0212     201      WRITE(6,106)
0213              WRITE(6,107)
         C WHICH SURFACE DO YOU WISH TO CONTOUR
0214              READ(IUT,101) ISUF
0215              IF(ISUF .LT. 1 .OR. ISUF .GT. 5) GO TO 201
0216              IF(ISUF .NE. 5) GO TO 215
0217              NSTOR =NPLNS(1) -1
0218              WRITE(6,113) NSTOR
         C WHICH KIND OF DIFFERENCE PLOT
0219              READ(IUT,101) IPLN
0220              INDX =1
```

```
0221                GO TO 202
0222      215       INDX =MAP(ISUF)
0223      202       WRITE(6,108)
          C WHAT CONTOUR LEVELS AND SPACING
0224                READ(IUT,109) BTLVL,TPLVL,SPAC
0225                IF(BTLVL .EQ. 0.) BTLVL =.01
0226                IF(TPLVL .EQ. 0.) TPLVL =.01
0227                IF(SPAC .EQ. 0.) GO TO 202
0228                NLV =(TPLVL -BTLVL)/SPAC +1
0229                IF(NLV .GT. 1000) GO TO 202
0230                MXDIF =0.
          C         DO 240 ILN =1,LINES-1
0231                DO 240 ILN=ISLNN,IELNN-1
0232                RS0 =(ILN-1)*12 +INDX
0233                RS1 =RS0 +12.
0234                RTP0 =(ILN-1)*12 +5
0235                RTP1 =RTP0 +12.
0236                IF(ISUF .NE. 5) GO TO 220
0237                IF(IPLN .GT. 0) GO TO 230
0238                CALL DPR(NEF,RS0,H(1,1),256.)
0239                CALL DPR(NEF,RS1,H(1,2),256.)
0240                GO TO 235
0241      230       RS0 =IPLN*200 +ILN +IFLD*20003.+2.
          C         CALL DPR(NER,RS0,H(1,1),256.)
          C         CALL DPR(NER,RS0+1.,H(1,2),256.)
0242      235       RS1 =ILN +IFLD*20003+2.
          C         CALL DPR(NER,RS1,H(1,3),256.)
          C         CALL DPR(NER,RS1+1.,H(1,4),256.)
0243                DO 236 II =1,ICOL
0244                I=II
0245                IF(MOD(ILN,2).EQ.0) I=ICOL-II+1
0246                H(I,1) =H(I,1) - H(I,3)
0247                H(I,2) =H(I,2) - H(I,4)
0248                IF(ABS(H(I,1)) .GT. MXDIF) MXDIF =ABS(H(I,1))
0249      236       CONTINUE
0250                GO TO 225
0251      220       CALL DPR(NEF,RS0,H(1,1),256.)
0252                CALL DPR(NEF,RS1,H(1,2),256.)
0253      225       CALL DPR(NEF,RTP0,NTP(1,1),256.)
0254                CALL DPR(NEF,RTP1,NTP1(1,1),256.)
          C         DO 240 IPZ =1,ICOL-1
          C         IP=IPZ
          C         IF(MOD(ILN,2).EQ.0) IP=ICOL-IPZ+1
0255                ISCL0=ISCL
0256                IECL0=IECL-1
0257                DO 240 IAQ=ISAQ,IEAQ
0258                XS=(IAQ-ISAQ)*(IECL-ISCL+1)
0259                DO 240 IPZ=ISCL0,IECL0
0260                IP=(IAQ-1)*NDSXX+IPZ
0261                IF(MOD(ILN,2).EQ.0)
                    1IP=(IEAQ-(IAQ-ISAQ))*NDSXX-(NDSXX-(IECL0-(IPZ-ISCL)))
0262                X=(XS+IPZ-ISCL+.5)*DH
0263                Y=(ILN-ISLNN+.5)*DH
0264                IF(MOD(ILN,2).EQ.0)
                    1 X=((IEAQ-IAQ)*(IECL-ISCL+1)+IECL0-IPZ)*DH+.5*DH
0265                NTPCP(1) =NTP(2,IP)
```

```
0266            NTPCR(2) =NTP(2,IP+1)
0267            NTPCR(3) =NTP1(2,IP+1)
0268            NTPCR(4) =NTP1(2,IP)
0269            NG =0
0270            DO 205 IN =1,4
0271            IF(NTPCR(IN) .EQ. 1) GO TO 204
0272            NG =NG +1
0273            NGA(IN) =1
0274            GO TO 205
0275    204     NGA(IN) =0
0276    205     CONTINUE
0277            IF(NG .LE. 2) GO TO 240
0278            HCR(1) =H(IP,1)
0279            HCR(2) =H(IP+1,1)
0280            HCR(3) =H(IP+1,2)
0281            HCR(4) =H(IP,2)
        C       X =(IP-1)*DH +DH/2.
        C       Y =(ILN-1)*DH +DH/2.
        D       WRITE(5,2050) IP,ILN,X,Y,HCR
        02050   FORMAT(2I10,2F10.2/4F10.2)
0282            HMIN=1.E8
0283            HMAX =-1.E8
0284            DO 210 I =1,4
0285            IF(NTPCR(I) .EQ. 1) GO TO 210
0286            IF(HCR(I) .GT. HMAX) HMAX =HCR(I)
0287            IF(HCR(I) .LT. HMIN) HMIN =HCR(I)
0288    210     CONTINUE
0289            IF(HMAX .LT. BTLVL .OR. HMIN .GT. TPLVL) GO TO 240
        D       WRITE(5,6777) BTLVL,TPLVL,SPAC,HMAX,HMIN
        06777   FORMAT(5F10.2)
0290            DO 241 CLVL =BTLVL,TPLVL,SPAC
        D       WRITE(5,6777) CLVL
0291            IF(CLVL .LE. HMAX .AND. CLVL .GE. HMIN)
                 1CALL SQUAR(X,Y,CLVL,HCR,NTPCR)
0292    241     CONTINUE
0293    240     CONTINUE
0294            IF(ISUF .EQ. 5) WRITE(6,114) MXOIF
0295            CALL PLTOUT
0296            GO TO 200
        C
        C               STREAM LINES
        C
        C DO YOU DESIRE STREAM LINES
0297    3000    POR=.1
0298            TMAX=1.E20
0299            WRITE(6,*) 'STREAMLINES AND TRAVEL TIMES 1=YES  CR=NO'
0300            READ(IUT,101) IC
0301            IF(IC.NE.1) GO TO 400
0302            WRITE(6,*)'DO YOU DESIRE TRAVEL TIMES 1=YES CR=JUST STREAMLINES'
0303            READ(IUT,101) ITVLTM
0304            IF(ITVLTM.NE.1) GO TO 3001
0305            WRITE(6,*) 'ENTER POROSITY 1F10.0'
0306            READ(IUT,*) POR
0307            WRITE(6,*) 'TO SET MAX. TRAVEL TIME ENTER A 1 FOR NONE A CR'
0308            READ(IUT,101) IS
0309            IF(IS.NE.1) GO TO 3001
```

```
0310            WRITE(6,*) 'ENTER MAX. TIME IN YEARS :F10.0'
0311            READ(IUT,*) TMAX
0312    3001    CONTINUE
0313    300     WRITE(6,*) 'MORE STREAMLINES OR TRAVEL TIMES 1=YES CR=NO'
0314            READ(IUT,101) IC
0315            IF(IC .EQ. 0) GO TO 400
0316            WRITE(6,130)
0317            READ(IUT,101) IM
0318            IF(IM .EQ. 1) GO TO 301
0319            WRITE(6,131)
0320            READ(IUT,132) IAQ,XL1,YL1,XL2,YL2,NLINE
0321            DST =SQRT((XL1-XL2)**2 +(YL1-YL2)**2)
0322            CNX =(XL2-XL1)/DST
0323            SNX =(YL2-YL1)/DST
0324            IF(NLINE .EQ. 1) NLINE =2
0325            IF(NLINE .GT. 50) NLINE =50
0326            DS =DST/(NLINE-1)
0327            GO TO 302
0328    301     WRITE(6,111)
        C STREAM LINE LOCATIONS
0329            READ(IUT,1099) IAQ,SX,SY
0330            NLINE=1
0331    1099    FORMAT(I5,2F10.0)
0332    302     XMIN=(IAQ-1)*NDSXX+ISCL
0333            XMAX=(IAQ-1)*NDSXX+IECL
0334            YMIN=ISLNN
0335            YMAX=IELNN
0336            WRITE(6,112)
        C UP OR DOWN GRADIENT
0337            READ(IUT,109) DR
0338            DO 325 IST=1,NLINE
0339            IF(IM.EQ.1) GO TO 303
0340            SX=(IST-1)*DS*CNX+XL1
0341            SY=(IST-1)*DS*SNX+YL1
0342    303     ICL0 =SX+(IAQ-1)*NDSXX
0343            IRW0 =SY
0344            X =SX+(IAQ-1)*NDSXX
0345            Y =SY
0346            ISTP =0
0347            TIME=0.
0348            YEARS=0.
0349            RB =(IRW0 -1)*12 +1
0350            CALL DPR(NEF,RB,H0,5,*256.)
0351            CALL DPR(NEF,RB+12.,H2,5,*256.)
0352            IPENPS=3
0353            SXX=SX-ISCL+1.+(IAQ-ISAQ)*(IECL-ISCL+1)
0354            SYY=SY-ISLNN+1.
0355            SXXX=SX+(IAQ-1)*NDSXX
0356            IF(SXXX.LT.XMIN.OR.SXXX.GT.XMAX.OR.SY.LT.YMIN.OR.SY.GT.YMAX)
                 1IPENPS=3
0357            IF(SXXX.LT.XMIN.OR.SXXX.GT.XMAX.OR.SY.LT.YMIN.OR.SY.GT.YMAX
                 1.AND.ITVLTM.EQ.1) GO TO 320
0358            IF(SXXX.LT.XMIN.OR.SXXX.GT.XMAX.OR.SY.LT.YMIN.OR.SY.GT.YMAX)
                 1 GO TO 6799
0359            CALL PLOTT((SXX-.5)*DH,(SYY-.5)*DH,IPENPS)
0360            IPENPS=2
```

```
0361    6799    CONTINUE
0362    305     CALL CORNR(X,Y,ICL0,IRW0,HCR,IDN,ISTP)
0363            IF(IDN .EQ. 1) GO TO 320
0364            A =HCR(3)-HCR(4)-HCR(2)+HCR(1)
0365            B =HCR(2) -HCR(1)
0366            C =HCR(4) -HCR(1)
0367    310     DX =X -ICL0
0368            DY =Y -IRW0
0369            GX =A*DY +B
0370            GY =A*DX +C
0371            GMAG =SQRT(GX*GX +GY*GY)
0372            IF(GMAG .EQ. 0.) GO TO 320
0373            IF(ISTP .EQ. 0) GO TO 315
0374            COSA =(GX*GXL + GY*GYL)/(GMAG*GMAGL)
0375            IF(COSA .LT. 0.) GO TO 320
0376    315     GXL =GX
0377            GYL =GY
0378            GMAGL =GMAG
0379            CN =GX/GMAG
0380            SN =GY/GMAG
0381            X =X +DR*DH*CN
0382            Y =Y +DR*DH*SN
0383            XX=X-(IAQ-1)*NDSXX-ISCL+1.+(IAQ-ISAQ)*(IECL-ISCL+1)
0384            YY=Y-ISLNN+1.
0385            IF(X.LT.XMIN.OR.X.GT.XMAX.OR.Y.LT.YMIN.OR.Y.GT.YMAX)
                1IPENPS=3
0386            IF(X.LT.XMIN.OR.X.GT.XMAX.OR.Y.LT.YMIN.OR.Y.GT.YMAX
                1.AND.ITVLTM.EQ.1) GO TO 320
0387            IF(X.LT.XMIN.OR.X.GT.XMAX.OR.Y.LT.YMIN.OR.Y.GT.YMAX)
                1GO TO 8333
0388            CALL PLOTT((XX-.5)*DH,(YY-.5)*DH,IPENPS)
0389            IPENPS=2
0390    8333    CONTINUE
0391            IF(ITVLTM.NE.1) GO TO 7333
0392            V=KAV*GMAG/(POR*DELX)
0393            TIME=TIME+ABS(DR*DH*DELX/V)
0394            YEARS=TIME/365.25
0395            IF(YEARS.GT.THAX) GO TO 320
0396    7333    CALL PLTOUT
0397            ISTP =ISTP +1
0398            IF(ISTP .GT. 500) GO TO 320
0399            GO TO 305
0400    320     CONTINUE
0401            IF(ITVLTM.NE.1) GO TO 324
0402            STREAM(1,IST) =X-(IAQ-1)*NDSXX
0403            STREAM(2,IST) =Y
0404            STREAM(3,IST) =YEARS
0405            STREAM(4,IST) =ISTP*ABS(DR)*DELX*DH
0406            WRITE(6,*) 'AQUIFER',IAQ
0407            WRITE(6,147) YEARS,STREAM(4,IST)
0408    324     CALL PLTOUT
0409    325     CONTINUE
0410            IF(ITVLTM.NE.1) GO TO 300
0411            WRITE(6,133)
0412            READ(IUT,101) IP
0413            IF(IP .EQ. 0) GO TO 300
```

```
0414              WRITE(5,134)
0415              WRITE(5,137)
0416              WRITE(5,141) IAQ
0417     141      FORMAT(10X,'AQUIFER',I3//)
0418              WRITE(5,140) POR
0419              IF(IS.EQ.1) WRITE(5,142) TMAX
0420              IF(IM.EQ.1) WRITE(5,1399) SX,SY
0421              IF(IM.NE.1) WRITE(5,139) XL1,YL1,XL2,YL2
0422              WRITE(5,136)
0423              SUM1=0.
0424              SUM2=0.
0425              DO 335 IST =1,NLINE
0426              WRITE(5,135)IST,(STREAM(J,IST),J =1,4)
0427              SUM1 =SUM1 +STREAM(3,IST)
0428              SUM2 =SUM2 +STREAM(4,IST)
0429     335      CONTINUE
0430              TAV =SUM1/NLINE
0431              DAV =SUM2/NLINE
0432              VAV=DAV/TAV
0433              WRITE(5,138) TAV,DAV,VAV
0434              CLOSE(UNIT=5)
0435              GO TO 300
0436     400      CALL PLOT(DH/2.,DH/2.,3)
0437              CALL NEWPEN(1)
0438              CALL PLOTND
0439              CLOSE(UNIT=7)
         C CROSSECTION PLOTS NOW?
0440              WRITE(6,*) 'CROSSECTION PLOTS NOW CR=NO 1=YES'
0441              READ(IUT,101) ICRS
0442              IF(ICRS.EQ.0) STOP
0443              IF(IDEV.EQ.0.OR.IDEV.EQ.2) GO TO 4440
0444              CALL PLOTS(0.,0.,7)
0445              CALL FACTOR(SF)
0446              CALL PLOT(WD+2.,0.,3)
0447              CALL PLOTND
0448              CLOSE(UNIT=7)
0449     4440     CONTINUE
0450              IFST=0
0451              IF(IMNU.EQ.1) GO TO 4993
         C
         C                CROSS SECTION PLOTS
         C
0452     500      IUT=6
0453              IFST=0
0454              WRITE(6,*) 'CROSSECTION PLOTS'
0455              WRITE(6,*) 'ENTER A 1 FOR NODE SYMBOLS 0 FOR NONE'
0456              READ(6,101) ISMOK
0457              WRITE(6,4999)
0458     4999     FORMAT(' MANUAL = CR OR AUTOMATIC =1')
0459              READ(6,101) IMNU
0460              IUT=6
0461              IF(IMNU.NE.1) GO TO 83290
0462              IUT=8
0463              WRITE(6,*) 'ENTER A NAME FOR THE AUTOMATIC CONTROL FILE'
0464              READ(6,1382) NAMM
0465              N=ICHR(NAMM)
```

```
0466              OPEN(UNIT=8,NAME=NAMM,TYPE='OLD',
                  1READONLY)
0467     83290    CONTINUE
0468              IF(IDEV.NE.2) GO TO 13860
0469              WRITE(6,*) 'ENTER THE NAME OF THE DISK PLOT FILE'
0470              READ(6,1382) NAMM
0471              N=ICHR(NAMM)
0472              OPEN(UNIT=7,NAME=NAMM,FORM='UNFORMATTED'
                  1,TYPE='NEW')
0473     13860    CONTINUE
0474     4993     YTD=0.
0475              XTD=0.
0476     5000     IF(IDEV.EQ.0) OPEN(UNIT=7,NAME='CROSS.PLT',FORM='UNFORMATTED'
                  1,TYPE='NEW')
0477              WRITE(6,122)
0478              READ(IUT,101) IOP
0479              IF(IOP .GT. 1) IOP =3
0480              WRITE(6,115)
0481              READ(IUT,109) SF
0482              IF(IDEV.EQ.0.OR.IFST.EQ.0) CALL PLOTS(0.,0.,7)
0483              IFST=1
0484              CALL NEWPEN(2)
0485              CALL FACTOR(SF)
0486     501      WRITE(6,118)
0487              READ(IUT,101) IAGNM,IX00,IY0,IX10,IY1
0488              FTIN =2000.
0489              IX0=IX00+(IAGNM-1)*NDSXX
0490              IX1=IX10+(IAGNM-1)*NDSXX
0491              WRITE(6,119)
0492              READ(IUT,101) ISF
0493              IF(ISF .EQ. 1) WRITE(6,120)
0494              IF(ISF .EQ. 1) READ(IUT,109) FTIN
0495              XDST =(IX1 -IX0)*DELX
0496              YDST =(IY1 -IY0)*DELX
0497              DTOT =SQRT(XDST*XDST +YDST*YDST)
0498              DS =DELX/2.
0499              NDS =DTOT/DS
0500              IF(NDS .GT. 300) NDS =300
0501              X0 =IX0*DELX
0502              Y0 =IY0*DELX
0503              CSN =XDST/DTOT
0504              SGN =YDST/DTOT
0505              ILNL =Y0/DELX
0506              DO 510 IP =1,NDS+1
0507              ILN =Y0/DELX
0508              ICL =X0/DELX
0509              IF(ILN .EQ. ILNL .AND. IP .NE. 1) GO TO 505
0510              RB =(ILN-1)*12. +1.
0511              CALL DPR(NEF,RB,H0,8.*256.)
0512              CALL DPR(NEF,RB+12.,H2,8.*256.)
0513              ILNL =ILN
0514     505      DO 508 IAG =1,IOP
0515              INDXA =MAPC(IAG)
0516              HCR(1) =HAG(ICL,INDXA,2)
0517              HCR(2) =HAG(ICL+1,INDXA,2)
0518              HCR(3) =HAG(ICL+1,INDXA,1)
```

```
0519              HCR(4) =HAQ(ICL,INDXA,1)
0520              NTPCR(1) =NTP(2,ICL)
0521              NTPCR(2) =NTP(2,ICL+1)
0522              NTPCR(3) =NTP1(2,ICL+1)
0523              NTPCR(4) =NTP1(2,ICL)
0524              DX =(X0 -ICL*DELX)/DELX
0525              DY =(Y0 -ILN*DELX)/DELX
0526              CALL POTVAL(HCR,NTPCR,DX,DY,BIAS,VAL)
0527              PRFILE(IP,IAQ) =VAL
0528      508     CONTINUE
0529              X0 =X0 +DS*CSN
0530              Y0 =Y0 +DS*SGN
0531      510     CONTINUE
          C
          C                 GENERATE GRAPH
          C
0532              HMIN =1.E5
0533              HMAX =-1.E5
0534              DO 515 IAQ =1,IOP
0535              DO 515 IP =1,NDS
0536              IF(PRFILE(IP,IAQ) .EQ. -1.E6) GO TO 515
0537              IF(PRFILE(IP,IAQ) .GT. HMAX) HMAX =PRFILE(IP,IAQ)
0538              IF(PRFILE(IP,IAQ) .LT. HMIN) HMIN =PRFILE(IP,IAQ)
0539      515     CONTINUE
0540              YSCAL =20.
0541              IF(HMAX -HMIN .GT. 200.) YSCAL =50.
0542              YDIV =YSCAL/2.
0543              HLOW =(IFIX(HMIN/YDIV) -1)*YDIV
0544              HTOP =(IFIX(HMAX/YDIV) +1)*YDIV
0545              NDY =(HTOP -HLOW)/YDIV
0546              NDX =DTOT/FTIN +3
0547              TKL =.1
0548              RLNG =NDX
0549              RHI =NDY
0550              IF(NDY .GT. 4) GO TO 516
0551              NDY =4
0552              RHI =NDY
0553              HTOP =HLOW +4.*YDIV
0554      516     IF(SF*(RLNG+2.0/SF).GT.XTD) XTD=SF*(RLNG+2.0/SF)
0555              YTD=YTD+SF*(RHI/2.+2.0+1./SF)
0556              IF(YTD.LT.31.) GO TO 5155
0557              IF(IDEV.EQ.0) GO TO 5155
0558              CALL PLOT(XTD/SF,-(YTD-SF*(RHI/2.+2.0+1./SF))/SF,-3)
0559              XTD=SF*(RLNG+2.0/SF)
0560              YTD=SF*(RHI/2.+2.0+1.0/SF)
0561      5155    CONTINUE
0562              CALL PLOT(0.,0.,-3)
0563              IF(NDX .GT. 50) GO TO 517
0564              CALL AXIS1(0.,0.,1.,NDX,TKL,0.)
0565              CALL AXIS1(RLNG,0.,.5,NDY,TKL,90.)
0566              CALL AXIS1(RLNG,RHI/2.,1.,NDX,TKL,180.)
0567              CALL AXIS1(0.,RHI/2.,.5,NDY,TKL,270.)
0568              GO TO 518
0569      517     CALL PLOT(1.*NDX,0.,2)
0570              CALL PLOT(1.*NDX,.5*NDY,2)
0571              CALL PLOT(0.,.5*NDY,2)
```

```
0572              CALL PLOT(0.,0.,2)
0573              GO TO 5255
0574     518      YL =-.3
0575              DO 520 DD =0.,NDX*FTIN,FTIN
0576              XPL =DD/FTIN
0577              IF(DD .NE. 0.) XPL =XPL -.15
0578              CALL NUMBER(XPL,YL,.1,DD,0.,0)
0579     520      CONTINUE
0580              XL =-.4
0581              DO 525 PHI =HLOW,HTOP,YDIV
0582              HP =(PHI -HLOW)/YSCAL
0583              CALL NUMBER(XL,HP,.1,PHI,0.,0)
0584     525      CONTINUE
0585     5255     CONTINUE
0586              VTEX =FTIN/YSCAL
0587              XMSG =NDX/2 -.72
0588              CALL SYMBOL(XMSG,-.6,.1,'DISTANCE (FT.)',0.,14)
0589              XTITL =NDX/2 -1.08
0590              CALL SYMBOL(XTITL,-1.,.14,'CROSS SECTION PLOT',0.,18)
0591              CALL SYMBOL(XTITL-.6,-1.5,.14,'VERTICAL EXAGGERATION = ',0.,24)
0592              CALL NUMBER(999.,999.,.14,VTEX,0.,1)
0593              YMSG =NDY/4. -.72
0594              CALL SYMBOL(-.6,YMSG,.1,'POTENTIAL (FT.)',90.,15)
0595              IF(HLOW .GT. 0.) GO TO 527
0596              IF(HTOP .LE. 0.) GO TO 527
0597              H00 =(0. -HLOW)/YSCAL
0598              CALL PLOT(0.,H00,3)
0599              CALL PLOT(RLNG-2.,H00,2)
0600     527      IF(IROTT .EQ. 1) GO TO 528
0601              XST =XLLC +(IX00-1)*DELX
0602              YST =YLLC +(IY0-1)*DELX
0603              XND =XLLC +(IX10-1)*DELX
0604              YND =YLLC +(IY1-1)*DELX
0605              GO TO 529
0606     528      XST =XLLC +(IY0-1)*DELX
0607              YST =YLLC +(NDSXX-IX00)*DELX
0608              XND =XLLC +(IY1-1)*DELX
0609              YND =YLLC +(NDSXX-IX10)*DELX
0610     529      XPLT =RLNG -1.75
0611              YPLT =(RHI -1.)/2.
0612              CALL SYMBOL(XPLT,YPLT,.1,'STARTING POINT',0.,14)
0613              CALL SYMBOL(XPLT,YPLT-.15,.1,'X0(FT) =',0.,8)
0614              CALL NUMBER(999.,999.,.1,XST,0.,0)
0615              CALL SYMBOL(XPLT,YPLT-.3,.1,'Y0(FT) =',0.,8)
0616              CALL NUMBER(999.,999.,.1,YST,0.,0)
0617              YPLT =YPLT -.5
0618              CALL SYMBOL(XPLT,YPLT,.1,'ENDING POINT',0.,12)
0619              CALL SYMBOL(XPLT,YPLT-.15,.1,'X1(FT) =',0.,8)
0620              CALL NUMBER(999.,999.,.1,XND,0.,0)
0621              CALL SYMBOL(XPLT,YPLT-.3,.1,'Y1(FT) =',0.,8)
0622              CALL NUMBER(999.,999.,.1,YND,0.,0)
0623              YPLT=YPLT-.5
0624              CALL SYMBOL(XPLT,YPLT,.1,'AQUIFER NO. =',0.,13)
0625              CALL NUMBER(999.,999.,.1,1.*IAQNM,0.,0)
0626              SIZDSM=.10
0627              SUN=.1*3F
```

```
0628              IF(SMN.GT..1) SIZDSM=.1/SF
0629              IF(ISMOK.EQ.0) SIZDSM=0.
0630              DO 530 IAQF =1,IOP
0631              CALL NEWPEN(2)
0632              H1 =(PRFILE(1,IAQF) -HLOW)/YSCAL
0633              IF(PRFILE(1,IAQF) .EQ. -1.E6) H1 =0.
0634              IF(IAQF.NE.2) CALL SYMBOL(0.,H1,SIZDSM,IAQF,0.,-1)
0635              IF(IAQF.EQ.2) CALL PLOT(0.,H1,3)
0636              DO 530 IP =1,NDS
0637              XD =(IP-1)*DS/FTIN
0638              YP =(PRFILE(IP,IAQF) -HLOW)/YSCAL
0639              IF(PRFILE(IP,IAQF) .EQ. -1.E6) YP =0.
0640              IF(IAQF.EQ.2) CALL PLOT(XD,YP,2)
0641              IF(IAQF.NE.2) CALL SYMBOL(XD,YP,SIZDSM,IAQF,0.,-2)
0642      530     CONTINUE
0643              CALL PLOT(0.,0.,3)
0644              CALL NEWPEN(2)
0645              CALL PLOT(0.,RHI/2.+1.0/SF+2.0,-3)
0646              CALL PLTOUT
0647              IF(IDEV.EQ.0) CALL PLOTND
0648              IF(IDEV.EQ.0) CLOSE(UNIT=7)
0649              WRITE(6,121)
0650              READ(IUT,101) IP
0651              IF(IP .EQ. 1) GO TO 5000
0652              CALL PLOTND
0653              CLOSE(UNIT=7)
0654              STOP
0655      100     FORMAT(1X,'ROTATION :0,1 ?')
0656      101     FORMAT(5I6)
0657      102     FORMAT(1X,'NODE TYPE PLOT ?')
0658      103     FORMAT(1X,'EXT  BNDY  HELD  INT  ND.NOS.')
0659      104     FORMAT(1X,' 1    2     3     4    5')
0660      105     FORMAT(1X,'CONTOUR PLOT ?')
0661      106     FORMAT(1X,' H  K  TOP  BOT  DIFF')
0662      107     FORMAT(1X,' 1  2   3    4    5')
0663      108     FORMAT(1X,'BOT LVL ,TOP LVL ,SPAC :3F6')
0664      109     FORMAT(3F8.0)
0665      111     FORMAT(1X,'START LOC. AQUIFER NO.  X-COLL ,Y-ROW :I5,2F10.0')
0666      112     FORMAT(1X,'UP GRAD ,DOWN GRAD :1,-1 F4.')
0667      113     FORMAT(1X,'H(N) - H(0) ,N = 1 TO',I3,' :I4 (CR=CURRENT H)')
0668      114     FORMAT(1X,'MAX DIFFERENCE =',F8.2,' FT.')
0669      115     FORMAT(1X,'SCALE FACTOR ,:F10.0')
0670      116     FORMAT(1X,'LARGE OR SMALL FIELD ? :CR=LG,1=SM')
0671      117     FORMAT(1X,'CROSS SECTION PLOTS ? :1=Y,CR=N')
0672      118     FORMAT(1X,'INPUT AQUIFER NUMBER AND POINT PAIRS'
                  1/1X,'AQUIFER ,COLL=0 ,LINE0 ,COLL=1 ,LINE1
                  1 :5I3')
0673      119     FORMAT(1X,'NEW GRAPH SCALE ? :1=Y,CR=2000(FT/IN)')
0674      120     FORMAT(1X,'NEW SCALE (FT/IN) :F6')
0675      121     FORMAT(1X,'MORE PROFILES ? :1=Y,CR=NO')
0676      122     FORMAT(1X,'CROSS SECTION OPTIONS'/1X,'(1) PLOT POTENTIAL'
                  1/1X,'(2) PLOT POT. ,TOP ,BOT. :PENS 1,2,3')
0677      147     FORMAT(1X,'TRAVEL TIME(YEARS) =',F10.2,5X,'DISTANCE(FEET) ='
                  1,F10.0)
0678      130     FORMAT(1X,'INPUT MODE SELECTION :I1'/
                  11X,'(1) SINGLE STREAMLINE'/
```

```
                11X,'(2) MULTIPLE STREAMLINE')
0679    131     FORMAT(1X,'INPUT AQUIFER NUMBER AND END POINTS OF LINE SEGMENT'/
                11X,'AQ. # ,X1 ,Y1 ,X2 ,Y2 ,NLINES :I2,4F10.0,I4')
0680    132     FORMAT(I2,4F10.0,I4)
0681    133     FORMAT(1X,'PRINT DATA ON STREAMLINES ? :1=YES')
0682    134     FORMAT(1H1)
0683    135     FORMAT(1X,I2,5X,2F10.4,F10.2,F10.1)
0684    136     FORMAT(1X,'NO.',9X,'X-END',5X,'Y-END',5X,'TIME',6X,'DISTANCE'/
                18X,'    (NODE)     (NODE)   (YEARS)     (FEET)'/)
0685    137     FORMAT(10X,'STREAMLINE DATA'//)
0686    138     FORMAT(//1X,'AVERAGE TIME(YEARS) =',F10.2/
                11X,'AVERAGE DISTANCE(FEET) =',F10.0
                1/1X,'AVERAGE VELOCITY(FEET/YEAR) =',F12.4)
0687    139     FORMAT(1X,'ENDPOINTS OF STARTING BOUNDRY'/
                11X,'IN NODE COORDINATES'//
                11X,'X1 =',F10.4,5X,'Y1 =',F10.4/
                11X,'X2 =',F10.4,5X,'Y2 =',F10.4//)
0688    1399    FORMAT(1X,'ENDPOINTS OF STARTING BOUNDRY'/
                11X,'IN NODE COORDINATES'//
                11X,'X1 =',F10.4,5X,'Y1 =',F10.4)
0689    140     FORMAT(1X,'POROSITY =',F6.4)
0690    142     FORMAT(1X,'MAX. TIME(YEARS) =',F13.0/)
0691            END
```

```
0001            SUBROUTINE AXIS1(X0,Y0,DH,NDH,TKL,ANG)
0002            PI =3.14159265
0003            ANG1 =ANG*PI/180.
0004            SN =SIN(ANG1)
0005            CO =COS(ANG1)
0006            DX =DH*CO
0007            DY =DH*SN
0008            TKX =-TKL*SN
0009            TKY =TKL*CO
0010            X =X0
0011            Y =Y0
0012            CALL PLOT(X0,Y0,3)
0013            DO 10 I =1,NDH
0014            DO 5 J =1,10
0015            X =X +DX/10.
0016            Y =Y +DY/10.
0017            CALL PLOT(X,Y,2)
0018            CALL PLOT(X+TKX/2.,Y+TKY/2.,2)
0019      5     CALL PLOT(X,Y,2)
0020            CALL PLOT(X+TKX,Y+TKY,2)
0021      10    CALL PLOT(X,Y,2)
0022            RETURN
0023            END
```

```
0001            SUBROUTINE POTVAL(HCR,NTPCR,DX,DY,BIAS,VAL)
0002            DIMENSION HCR(4),NTPCR(4)
0003            VAL =-1.E6
0004            DO 10 I =1,4
0005            IF(NTPCR(I) .EQ. 1) RETURN
0006      10    CONTINUE
0007            A =HCR(2) -HCR(1)
0008            B =HCR(4) -HCR(1)
0009            C =HCR(1) +HCR(3) -HCR(2) -HCR(4)
0010            U =HCR(1) +A*DX +B*DY +C*DX*DY
0011            VAL =U -BIAS
0012            RETURN
0013            END
```

```
0001            SUBROUTINE CORNR(X,Y,ICL0,IRW0,HCR,IDN,ISTP)
0002            DIMENSION H(128,16),HCR(4),H0(128),H2(128)
0003            INTEGER NTP0(2,128),NTP2(2,128),NTPCR(4)
0004            REAL K0(128),K2(128),KAV
0005            COMMON /BK0/ H
0006            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,IROT
0007            COMMON /BLK1/ NODE,NEF,KAV
0008            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAOPOS,ICLMIN,ICLMAX
0009            EQUIVALENCE (H(1,1),H2),(H(1,9),H0),(H(1,5),NTP2),(H(1,13),NTP0)
                1,(H(1,3),K2),(H(1,11),K0)
0010            INS =0
0011            IDN =0
0012            ICL =X
0013            IRW =Y
0014            IF(IRW .LE. 0 .OR. IRW .GE. LINES) GO TO 20
0015            IF(ICL .LE. 0 .OR. ICL .GE. ICOL) GO TO 20
0016            IF(ISTP .EQ. 0) GO TO 11
0017            IF(ICL .EQ. ICL0 .AND. IRW .EQ. IRW0) RETURN
0018            IF(ICL .NE. ICL0) INS =1
0019            IF(IRW .EQ. IRW0) GO TO 10
0020            RB =(IRW-1)*12 +1
0021            CALL DPR(NEF,RB,H0,5.*256.)
0022            CALL DPR(NEF,RB+12.,H2,5.*256.)
0023            INS =1
0024      10    IF(INS .EQ. 0) RETURN
0025      11    NTPCR(1) =NTP0(2,ICL)
0026            NTPCR(2) =NTP0(2,ICL+1)
0027            NTPCR(3) =NTP2(2,ICL+1)
0028            NTPCR(4) =NTP2(2,ICL)
0029            ICL0 =ICL
0030            IRW0 =IRW
0031            DO 15 I =1,4
0032            IF(NTPCR(I) .EQ. 1) GO TO 20
0033      15    CONTINUE
0034            HCR(1) =H0(ICL)
0035            HCR(2) =H0(ICL+1)
0036            HCR(3) =H2(ICL+1)
0037            HCR(4) =H2(ICL)
0038            KAV=(K0(ICL)+K0(ICL+1)+K2(ICL)+K2(ICL+1))/4.
0039            RETURN
0040      20    IDN =1
0041            RETURN
0042            END
```

```
0001            SUBROUTINE PLTIT(X,Y,IST,KND)
0002            DIMENSION BOX(2,8)
0003            INTEGER IP(4)
0004            REAL KAV
0005            COMMON /BLK1/ NODE,NEF,KAV
0006            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0007            DATA BOX/1.,0.,1.,1.,0.,1.,-1.,1.,-1.,0.,-1.,-1.,0.,-1.,1.,-1./
0008            DHX =.475*DH
0009            IKND =KND
0010            IP(1) =IKND/1000
0011            IKND =IKND -IP(1)*1000
0012            IP(2) =IKND/100
0013            IKND =IKND -IP(2)*100
0014            IP(3) =IKND/10
0015            IKND =IKND -IP(3)*10
0016            IP(4) =IKND
0017            X0 =X
0018            Y0 =Y
0019            IF(NODE .EQ. 1) GO TO 15
0020            IF(IST .EQ. 0) GO TO 5
0021            X0 =X +DHX*BOX(1,IST)
0022            Y0 =Y +DHX*BOX(2,IST)
0023      5     CALL PLOTT(X0,Y0,3)
0024            DO 10 I =1,4
0025            IPT =IP(I)
0026            IF(IPT .EQ. 0) GO TO 10
0027            XP =DHX*BOX(1,IPT) +X
0028            YP =DHX*BOX(2,IPT) +Y
0029            CALL PLOTT(XP,YP,2)
0030      10    CONTINUE
0031            CALL PLOTT(X0,Y0,2)
0032            RETURN
0033      15    DHX =.5*DH
0034            IP1 =IP(1)
0035            IF(IP(1) .EQ. 0) IP1 =IP(2)
0036            IP2 =IP(4)
0037            XP =DHX*BOX(1,IP1) +X
0038            YP =DHX*BOX(2,IP1) +Y
0039            CALL PLOTT(XP,YP,3)
0040            CALL PLOTT(X0,Y0,2)
0041            XP1 =DHX*BOX(1,IP2) +X
0042            YP1 =DHX*BOX(2,IP2) +Y
0043            CALL PLOTT(XP1,YP1,2)
0044            RETURN
0045            END
```

```
0001            SUBROUTINE PLOTT(X,Y,IK)
0002            COMMON /HEAD/ N,M,LINES,ICOL,NIT,ITCT,IROT
0003            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0004            IF(IROT .GE. 1) GO TO 5
0005            CALL PLOT(X,Y,IK)
0006            RETURN
0007      5     YP =ICOLN*DH -X
0008            XP =Y
0009            CALL PLOT(XP,YP,IK)
0010            RETURN
0011            END
```

```
0001            SUBROUTINE SQUAR(X,Y,CLVL,HCR,NTPCR)
0002            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0003            DIMENSION HCR(4),NTPCR(4),CP(2,4),SQPT(2,4)
0004            IF(NG .EQ. 4) GO TO 40
0005            DO 5 I =1,4
0006            IF(NGA(I) .EQ. 0) GO TO 6
0007     5      CONTINUE
0008            RETURN
0009     6      I1 =I-1
0010            I2 =I+1
0011            IF(I1 .EQ. 0) I1 =4
0012            IF(I2 .EQ. 5) I2 =1
0013            H5 =(HCR(I1) +HCR(I2))/2.
0014    10      CONTINUE
0015    15      GO TO (20,25,30,35),I
0016    20      CALL TRIPLT(2,X,Y,HCR(2),HCR(3),H5,CLVL)
0017            CALL TRIPLT(3,X,Y,HCR(3),HCR(4),H5,CLVL)
0018            RETURN
0019    25      CALL TRIPLT(3,X,Y,HCR(3),HCR(4),H5,CLVL)
0020            CALL TRIPLT(4,X,Y,HCR(4),HCR(1),H5,CLVL)
0021            RETURN
0022    30      CALL TRIPLT(4,X,Y,HCR(4),HCR(1),H5,CLVL)
0023            CALL TRIPLT(1,X,Y,HCR(1),HCR(2),H5,CLVL)
0024            RETURN
0025    35      CALL TRIPLT(1,X,Y,HCR(1),HCR(2),H5,CLVL)
0026            CALL TRIPLT(2,X,Y,HCR(2),HCR(3),H5,CLVL)
0027            RETURN
0028    40      A =(HCR(1) +HCR(3) -HCR(2) -HCR(4))/(DH*DH)
0029            B =(HCR(2) -HCR(1))/DH
0030            C =(HCR(4) -HCR(1))/DH
0031            D =HCR(1)
0032            IF(ABS(A) .LT. 1.E-8) GO TO 41
0033            R =(SQRT(B*B +C*C)/ABS(A))/DH
0034            GO TO 42
0035    41      R =1.E6
0036    42      NCP =0
0037            CALL CROSS(HCR(1),HCR(2),CLVL,FR,ICRS)
0038            IF(ICRS .EQ. 0) GO TO 45
0039            NCP =NCP +1
0040            CP(1,NCP) =X +FR*DH
0041            CP(2,NCP) =Y
0042    45      CALL CROSS(HCR(2),HCR(3),CLVL,FR,ICRS)
0043            IF(ICRS .EQ. 0) GO TO 50
0044            NCP =NCP +1
0045            CP(1,NCP) =X +DH
0046            CP(2,NCP) =Y +FR*DH
0047    50      CALL CROSS(HCR(4),HCR(3),CLVL,FR,ICRS)
0048            IF(ICRS .EQ. 0) GO TO 55
0049            NCP =NCP +1
0050            CP(1,NCP) =X +FR*DH
0051            CP(2,NCP) =Y +DH
0052    55      CALL CROSS(HCR(1),HCR(4),CLVL,FR,ICRS)
0053            IF(ICRS .EQ. 0) GO TO 60
0054            NCP =NCP +1
```

```
0055                CP(1,NCP) =X
0056                CP(2,NCP) =Y +FR*DH
0057      60        CONTINUE
          D         WRITE(5,6066) CLVL,NCP,R,CP
          D6066     FORMAT(F10.2,I10,F10.2/4(2F10.4,5X))
0058                IF(NCP.LT.2) RETURN
0059                IF(NCP .EQ. 4) GO TO 65
0060                IF(R .LT. 10.) GO TO 70
0061                CALL PLOTT(CP(1,1),CP(2,1),3)
0062                CALL PLOTT(CP(1,2),CP(2,2),2)
0063                RETURN
0064      65        CALL PLOTT(CP(1,1),CP(2,1),3)
0065                CALL PLOTT(CP(1,3),CP(2,3),2)
0066                CALL PLOTT(CP(1,2),CP(2,2),3)
0067                CALL PLOTT(CP(1,4),CP(2,4),2)
0068                RETURN
0069      70        X0 =CP(1,1) -X
0070                Y0 =CP(2,1) -Y
0071                CALL PLOTT(CP(1,1),CP(2,1),3)
0072                X1 =CP(1,2) -X
0073                Y1 =CP(2,2) -Y
0074                DD =SQRT((X1-X0)**2 +(Y1-Y0)**2)
0075                IF(DD .LT. DH/5.) GO TO 100
0076                CALL SQPTS(X0,Y0,DH,A,B,C,D,CLVL,SQPT,ICP)
0077                XT =SQPT(1,1)
0078                YT =SQPT(2,1)
0079                IF(XT .LT. 0 .OR. XT .GT. DH) GO TO 72
0080                IF(YT .LT. 0 .OR. YT .GT. DH) GO TO 72
0081                XP =XT
0082                YP =YT
0083                GO TO 74
0084      72        XP =SQPT(1,2)
0085                YP =SQPT(2,2)
0086      74        CALL PLOTT(XP+X,YP+Y,2)
0087      75        CALL SQPTS(XP,YP,DH,A,B,C,D,CLVL,SQPT,ICP)
0088                XS1 =SQPT(1,1)
0089                YS1 =SQPT(2,1)
0090                XS2 =SQPT(1,2)
0091                YS2 =SQPT(2,2)
0092                DD1 =SQRT((XS1-X0)**2 +(YS1-Y0)**2)
0093                DD2 =SQRT((XS2-X0)**2 +(YS2-Y0)**2)
0094                IF(DD1 .GT. DD2) GO TO 80
0095                IF(DD2 .GT. DD) GO TO 100
0096                CALL PLOTT(XS2+X,YS2+Y,2)
0097                XP =XS2
0098                YP =YS2
0099                GO TO 75
0100      80        IF(DD1 .GT. DD) GO TO 100
0101                CALL PLOTT(XS1+X,YS1+Y,2)
0102                XP =XS1
0103                YP =YS1
0104                GO TO 75
0105      100       CALL PLOTT(X1+X,Y1+Y,2)
0106                RETURN
0107                END
```

```
0001              SUBROUTINE SQPTS(X,Y,DH,A,B,C,D,CL,SQPT,ICP)
0002              DIMENSION SQPT(2,4)
0003              ICP =0
0004              DS =DH/20.
0005              YA =Y -DS
0006              YB =Y +DS
0007              XA =X -DS
0008              XB =X +DS
0009              X1 =(CL -D -C*YA)/(A*YA +B)
0010              IF(X1 .LT. XA .OR. X1 .GT. XB) GO TO 10
0011              ICP =ICP +1
0012              SQPT(1,ICP) =X1
0013              SQPT(2,ICP) =YA
0014     10       Y1 =(CL -D -B*XB)/(A*XB +C)
0015              IF(Y1 .LT. YA .OR. Y1 .GT. YB) GO TO 20
0016              ICP =ICP +1
0017              SQPT(1,ICP) =XB
0018              SQPT(2,ICP) =Y1
0019     20       X2 =(CL -D -C*YB)/(A*YB +B)
0020              IF(X2 .LT. XA .OR. X2 .GT. XB) GO TO 30
0021              ICP =ICP +1
0022              SQPT(1,ICP) =X2
0023              SQPT(2,ICP) =YB
0024     30       Y2 =(CL -D -B*XA)/(A*XA +C)
0025              IF(Y2 .LT. YA .OR. Y2 .GT. YB) GO TO 40
0026              ICP =ICP +1
0027              SQPT(1,ICP) =XA
0028              SQPT(2,ICP) =Y2
0029     40       RETURN
0030              END
```

```
0001          SUBROUTINE CROSS(H1,H2,CLVL,FR,ICRS)
0002          COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
              1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0003          ICRS =0
0004          FR =0.
0005          IF(H1 .EQ. H2) RETURN
0006          ABSL =ABS(H1 -CLVL) +ABS(H2 -CLVL)
0007          IF(ABSL .GT. ABS(H2 -H1)) RETURN
0008          FR =(CLVL -H1)/(H2 -H1)
0009          ICRS =1
0010          RETURN
0011          END
```

```
0001            SUBROUTINE TRIPLT(NCR,X,Y,HA,HB,HC,CLVL)
0002            COMMON /BLK/ DH,NGA(4),NG,ICOLN,LINESN,ISAQ,IEAQ,ISCL,IECL
                1,ISLNN,IELNN,IAQPOS,ICLMIN,ICLMAX
0003            DIMENSION XYCR(2,3),XYREL(6,4),CRSPT(2,3)
0004            DATA XYREL/0.,0.,1.,0.,.5,.5,1.,0.,1.,1.,.5,.5
                1,1.,1.,0.,1.,.5,.5,0.,1.,0.,0.,.5,.5/
0005            DO 5 I =1,3
0006            XYCR(1,I) =X +DH*XYREL(2*I-1,NCR)
0007            XYCR(2,I) =Y +DH*XYREL(2*I,NCR)
0008      5     CONTINUE
0009            ICP =0
0010            CALL CROSS(HA,HB,CLVL,FR,ICRS)
0011            IF(ICRS .EQ. 0) GO TO 10
0012            ICP =ICP +1
0013            CRSPT(1,ICP) =XYCR(1,1) +FR*(XYCR(1,2) -XYCR(1,1))
0014            CRSPT(2,ICP) =XYCR(2,1) +FR*(XYCR(2,2) -XYCR(2,1))
0015     10     CALL CROSS(HB,HC,CLVL,FR,ICRS)
0016            IF(ICRS .EQ. 0) GO TO 15
0017            ICP =ICP +1
0018            CRSPT(1,ICP) =XYCR(1,2) +FR*(XYCR(1,3) -XYCR(1,2))
0019            CRSPT(2,ICP) =XYCR(2,2) +FR*(XYCR(2,3) -XYCR(2,2))
0020     15     CALL CROSS(HC,HA,CLVL,FR,ICRS)
0021            IF(ICRS .EQ. 0) GO TO 20
0022            ICP =ICP +1
0023            CRSPT(1,ICP) =XYCR(1,3) +FR*(XYCR(1,1) -XYCR(1,3))
0024            CRSPT(2,ICP) =XYCR(2,3) +FR*(XYCR(2,1) -XYCR(2,3))
0025     20     IF(ICP .NE. 2) RETURN
0026            CALL PLOTT(CRSPT(1,1),CRSPT(2,1),3)
0027            CALL PLOTT(CRSPT(1,2),CRSPT(2,2),2)
0028            RETURN
0029            END
```

```
DB0:CALTYP,LP/SH=CALTYP/CC,[1,1]BNWLIB,OLB/LB
/
UNITS=6
ASG=DP0:1
ASG=XY:7
ASG=TG:8
ASG=TI:2
MAXBUF=800
/
```

```
      C        ******CALTYP******
      C PROGRAM TO PLOT OUT THE CALCULATIONAL TYPES FOR THE MULTAQUIFER MODEL
0001            BYTE IWH
0002            COMMON/BLK/SCALEF,KOUNT,INNOS
0003          REAL IS
0004          DIMENSION TITLE(20),KEY(360),KTY(360), FILE(7)
      C         READ FILE NAME FROM TELETYPE
0005            KCHK=0
0006            IMAX=0
0007            WRITE(2,100)
0008    100     FORMAT(' ENTER NAME OF CALCULATION TYPE FILE (7A4)' )
0009            READ(2,110) FILE
0010    110     FORMAT(7A4)
0011            N=ICHR(FILE)
0012            OPEN(UNIT=1,NAME=FILE,TYPE='OLD',FORM='FORMATTED',READONLY)
0013            WRITE(2,101)
0014    101     FORMAT(' OUTPUT TO XY (X) , OUTPUT TO TG (T)')
0015            READ(2,102) IWH
0016    102     FORMAT(A1)
0017            IF(IWH.EQ.'X') CALL PLOTS(0.,0.,7)
0018            IF(IWH.EQ.'T') CALL PLOTS(0.,0.,8)
0019            WRITE(2,103)
0020    103     FORMAT(' LABEL CALCULATIONAL TYPES (CR) , LABEL NODE NOS. (1)')
0021            READ(2,104) INNOS
0022    104     FORMAT(I5)
0023            IFGGG=0
0024          READ(1,1) MAXX,MAXY,MAXA
0025    1     FORMAT(3I5)
0026          XMAX=FLOAT(MAXX+1)
0027          YMAX=FLOAT(MAXY+1)
0028          READ(1,2)  SCALEF
0029          CALL FACTOR(SCALEF)
0030    2     FORMAT(F10.0)
0031          X0=0.0
0032          Y0=-YMAX-2.
0033          DO 3 IAN=1,MAXA
0034          CALL NEWPEN(2)
0035            KOUNT=0
0036          Y0=Y0+YMAX+2.
0037          Y=Y0*SCALEF
0038          IF(Y+SCALEF*FLOAT(MAXY+1).GT.33.)  X0=X0+FLOAT(MAXX+2)
0039          IF(Y+SCALEF*FLOAT(MAXY+1).GT.33.)  Y0=0.0
0040          YX=YMAX+2.
0041          IF(Y0.EQ.0.)YX=0.
0042          CALL PLOT(X0,YX,-3)
0043          X0=0.0
      C DRAW BOX
0044          CALL PLOT(0.  ,0.  ,2)
0045          CALL PLOT(0.  ,YMAX,2)
0046          CALL PLOT(XMAX,YMAX,2)
0047          CALL PLOT(XMAX,  0.,2)
0048          CALL PLOT(  0.,  0.,2)
0049            DO 1000 IX=1,MAXX
0050            R=IX
0051            Y=-.5
0052            NDIGIT=1
```

```
0053              IF(R.GT.10.) NDIGIT=2
0054              IF(R.GT.100.)NDIGIT=3
0055              X=1.0+(IX-1)*1.0-6./7.*NDIGIT*.25/2.
0056              CALL NUMBER(X,Y,.25,R,0.,-1)
0057     1000     CONTINUE
0058              DO 200 IY=1,MAXY
0059              R=IY
0060              Y=1.0+(IY-1)*1.0-.25/2.
0061              NDIGIT=1
0062              IF(R.GT.10.)NDIGIT=2
0063              IF(R.GT.100.)NDIGIT=3
0064              X=-.5-6./7.*NDIGIT*.25/2.
0065              CALL NUMBER(X,Y,.25,R,0.,-1)
0066     200      CONTINUE
0067            CALL PLOT(  0.,  0.,-3)
0068              IF(IFGGG.NE.0) READ(1,40)
0069              IF(IFGGG.NE.0) READ(1,40)
0070              IFGGG=1
0071            READ(1,40) TITLE
0072     40     FORMAT(20A4)
0073            IS=.4
0074            IF(80.*IS.GT.XMAX-3.)  IS=(XMAX-3.)/80.
0075            CALL NEWPEN(2)
0076            CALL SYMBOL(3.,0.1,IS,TITLE,0.,80)
0077            CALL NEWPEN(2)
0078              ICOLS=1
0079              ICOLSP=MAXX
0080              ICLINC=1
0081     41     CONTINUE
0082            READ(1,4) IB,IT,KKK
0083     4      FORMAT(3I3)
         0        WRITE(2,4)IB,IT,KKK
0084            READ(1,5) ( KEY(JJJ),KTY(JJJ),JJJ=1,KKK)
0085     5      FORMAT(14(I3,O2))
0086            KEY(KKK+1)=0
0087            DO 9 ILIN=IB,IT
0088              JINC=1
0089            JJJ=1
0090              IF(ICOLS.EQ.MAXX)JJJ=KKK
0091              IF(ICOLS.EQ.MAXX)JINC=-1
0092              DO 8 ICOL=ICOLS,ICOLSP,ICLINC
0093              IF(JINC.EQ.-1)GO TO 60
0094            IF(ICOL -KEY(JJJ+1)) 7,6,7
0095     6      JJJ=JJJ+JINC
0096              GO TO 7
0097     60       IF(ICOL.GE.KEY(JJJ))GO TO 7
0098              JJJ=JJJ+JINC
0099     7      CONTINUE
0100            CALL NODE(ILIN,ICOL,KTY(JJJ))
0101     8      CONTINUE
0102              ICOL1=1
0103              ICOL2=MAXX
0104              ICLINC=1
0105              IF(ICOLS.EQ.1)ICOL2=1
0106              IF(ICOLS.EQ.1)ICOL1=MAXX
0107              IF(ICOLS.EQ.1) ICLINC=-1
0108              ICOLS=ICOL1
0109              ICOLSP=ICOL2
0110     9      CONTINUE
0111            IF(IT.LT.MAXY)  GO TO 41
0112     3      CONTINUE
0113            CALL PLOTNO
0114              CLOSE(UNIT=7)
0115              STOP
0116            END
```

```
0001          SUBROUTINE NODE(IL,IC,ITYPE)
0002            COMMON/BLK/SCALEF,KOUNT,INNOS
0003          DIMENSION GL(5)
0004          DIMENSION ONE0(12),ONE1(10),TWO(10),THREE(12),DOIT(12)
0005          DATA  ONE0/0.,0., .5,0., .5,-.5, -.5,-.5, -.5,0., 0.,0./
0006          DATA  ONE1/0.,0., .5,-.5, -.5,-.5, -.5,.5, 0.,0./
0007          DATA  TWO/0.,0., .5,-.5, -.5,-.5, -.5,0., 0.,0./
0008          DATA  THREE/0.,0., .5,-.5, .5,.5, -.5,.5, -.5,0., 0.,0./
0009          IKIND=IFLD(3,2,ITYPE)
0010          IROT=IFLD(0,3,ITYPE)
0011          IF(IKIND.EQ.0) GO TO 400
0012          IKIND=IKIND
0013          GO TO (100,200,300),IKIND
0014    100   IGO=IROT+1
0015          GO TO (10,11,10,11,10,11,10,11 ),IGO
0016    10    NO=6
0017          CALL ROT(ONE0,IROT,6,DOIT)
0018    400   XREF=FLOAT(IC)
0019          YREF=FLOAT(IL)
0020          SCALR=0.95
0021            IF(IKIND .NE.0)GO TO 82
0022            IS=14
0023            NO=1
0024            DOIT(1)=0.
0025            DOIT(2)=0.
0026            IPLACE=IROT+1
0027            GO TO (410,420,430),IPLACE
0028    410     IS=2
0029            CALL NEWPEN(2)
0030            GO TO 82
0031    420     IS=2
0032    430     CALL NEWPEN(2)
0033    82      ZR=0.0
0034          IZ=0
0035          QT=0.20
0036          M1=-1
0037          IF(IKIND.EQ.0.AND.IROT.EQ.2)  GO TO 84
0038          CALL SYMBOL(XREF,YREF,QT,IS,ZR,M1)
0039    84    CONTINUE
0040          IF(IKIND.EQ.0.AND.IROT.EQ.0) GO TO 999
0041          IF(IKIND.EQ.0.AND.IROT.EQ.1) RETURN
0042          CALL NEWPEN(2)
0043            IF(SCALEF.LE.0.24)GO TO 830
0044          IB=10*IKIND
0045          IB=IB+IROT
0046          X=XREF-.20
0047          Y=YREF+.10
0048            IF(INNOS.NE.1) GO TO 8301
0049          X=XREF-.35
0050            IF(IKIND.EQ.0.AND.(IROT.EQ.1.OR.IROT.EQ.2)) GO TO 8300
0051            KOUNT=KOUNT+1
0052            F=FLOAT(KOUNT)
0053          CALL NUMBER(X,Y,QT,F,ZR,IZ)
0054            GO TO 8300
0055    8301    F=FLOAT(IB)
0056            CALL NUMBER(X,Y,QT,F,ZR,IZ)
```

```
0057    8300      X=XREF-.450
0058            Y=YREF-.25
0059            QT=.14
0060            ENCODE(7,83,GL) IC,IL
0061    83      FORMAT(1H(,I2,1H,I2,1H))
0062            CALL SYMBOL(X,Y,QT,GL,ZR,7)
0063    830       CONTINUE
0064            IF(IKIND.EQ.0.AND.IROT.EQ.2)  GO TO 999
        C
        C         PLOT CALCULATIONAL TYPE
        C
0065            CALL  PLOT(XREF, YREF,3)
0066            DO 401 IEM=1,ND
0067              NP=1
0068              IF(IEM.EQ.2.OR.IEM.EQ.ND)NP=2
0069              CALL NEWPEN(2)
0070            J=(IEM-1)*2+1
0071            X=XREF+DOIT(J)*SCALR
0072            Y=YREF+DOIT(J+1)*SCALR
0073            CALL PLOT(X,Y,2)
0074            CALL PLOT(X,Y,2)
0075    401     CONTINUE
0076            RETURN
0077    999     CONTINUE
0078              CALL NEWPEN(2)
0079              IF(ITYPE.EQ.2)GO TO 991
        C
        C         WRITE NO. OF ACTIVE NODE
        C
0080              CALL NEWPEN(2)
0081            X=XREF-.35
0082            Y=YREF+.10
0083              KOUNT=KOUNT+1
0084              F=FLOAT(KOUNT)
0085              IF(INNOS.EQ.1) CALL NUMBER(X,Y,QT,F,ZR,IZ)
        C
        C         PLOT BOX FOR TYPE 0 AND TYPE 2
        C
0086    991        CALL PLOT(XREF+.5*SCALR,YREF+.5*SCALR,3)
0087            CALL PLOT(XREF+.5*SCALR,YREF-.5*SCALR,2)
0088            CALL PLOT(XREF-.5*SCALR,YREF-.5*SCALR,2)
0089            CALL PLOT(XREF-.5*SCALR,YREF+.5*SCALR,2)
0090            CALL PLOT(XREF+.5*SCALR,YREF+.5*SCALR,2)
0091            IF(IROT.EQ.2)   GO TO 998
0092    997       RETURN
        C
        C         PLOT DIAGONALS FOR TYPE NO. 2
        C
0093    998     CONTINUE
0094            CALL PLOT(XREF-.5*SCALR,YREF-.5*SCALR,2)
0095            CALL PLOT(XREF+.5*SCALR,YREF-.5*SCALR,2)
0096            CALL PLOT(XREF-.5*SCALR,YREF+.5*SCALR,2)
0097            GO TO 997
0098    11      ND=5
0099            IGRG=-IROT
0100            CALL ROT(ONE1, IGRG,5,DOIT)
0101            GO TO 400
0102    200     ND=5
0103            CALL ROT(TWO,IROT,5,DOIT)
0104            GO TO 400
0105    300     ND=6
0106            CALL ROT(THREE,IROT,6,DOIT)
0107            GO TO 400
0108            END
```

```
0001          SUBROUTINE ROT(A,IROT,NO,OUT)
0002          DIMENSION OUT(12),A(12)
0003          IF(IROT.LT.0)  GO TO 900
0004          IGO=IROT+1
0005          GO TO (100,200,300,400,500,600,700,800),IGO
0006    100   DO 101 I=1,NO
0007          J=(I-1)*2+1
0008          OUT(J)=A(J)
0009    101   OUT(J+1)=A(J+1)
0010          IST=100
0011          RETURN
0012    200   DO 201 I=1,NO
0013          J=(I-1)*2+1
0014          OUT(J)=A(J+1)
0015    201   OUT(J+1)=A(J)
0016          IST=200
0017          RETURN
0018    300   DO 301 I=1,NO
0019          J=(I-1)*2+1
0020          OUT(J)=A(J+1)
0021    301   OUT(J+1)=-A(J)
0022          IST=300
0023          RETURN
0024    400   DO 401 I=1,NO
0025          J=(I-1)*2+1
0026          OUT(J)=A(J)
0027    401   OUT(J+1)=-A(J+1)
0028          IST=400
0029          RETURN
0030    500   DO 501 I=1,NO
0031          J=(I-1)*2+1
0032          OUT(J)=-A(J)
0033    501   OUT(J+1)=-A(J+1)
0034          IST=500
0035          RETURN
0036    600   DO 601 I=1,NO
0037          J=(I-1)*2+1
0038          OUT(J)=-A(J+1)
0039    601   OUT(J+1)=-A(J)
0040          IST=600
0041          RETURN
0042    700   DO 701 I=1,NO
0043          J=(I-1)*2+1
0044          OUT(J)=-A(J+1)
0045    701   OUT(J+1)=A(J)
0046          IST=700
0047          RETURN
0048    800   DO 801 I=1,NO
0049          J=(I-1)*2+1
0050          OUT(J)=-A(J)
0051    801   OUT(J+1)=A(J+1)
0052          IST=800
0053          RETURN
0054    900   IGO=-IROT+1
0055          GO TO (1000,1001,1002,1003,1004,1005,1006,1007),IGO
0056    1000  STOP 111
0057    1001  GO TO 100
0058    1003  GO TO 300
0059    1005  GO TO 500
0060    1007  GO TO 700
0061          END
```

```
DB0:DIFER,LP/SH=DIFER,:1,:1BNHLIB/LB
/
UNITS=7
ACTFIL=7
ASG=TI:6
ASG=DB0:1
ASG=DP0: 5:2:3:7
ASG=LP0:4.
/
```

THIS PAGE INTENTIONALLY
LEFT BLANK

DIFER

THIS PAGE INTENTIONALLY
LEFT BLANK

```
        C ***** DIFER ******
        C ***** PROGRAM TO MAKE DIFFERENCE MAPS FOR THE VTT PROGRAM
        C DIFER,LP/SH=DIFER,[1,1]BNWLIB/LB
        C /
        C UNITS=7
        C ACTFIL=7
        C ASG=TI:6
        C ASG=DP0: 5:2:3:7
        C ASG=LP0:4
        C /
        C
        C PROGRAM TO MAKE A DIFFERENCE SURFACE FOR PLOTTING
        C AND TO MAKE A PRINTOUT OF THE DIFFERENCE
        C
0001            BYTE A,BB,C,D,E,F,BLNK,OUT(128),INTH1(10),INTH2(206),INTH3(206)
                1,INTH4(206)
0002            BYTE DOIT,DOIT1,TITL
0003            INTEGER OXFER,TRNOSS,YEAR,DAY,HOUR,YEARL,DAYL,HOURL,AUTO
        C
0004            DIMENSION BL(2,256),B(256),BASE(256),NEW(256),FIL1(7),FIL2(7)
0005            DIMENSION BV(5),IS(5)
0006            DIMENSION FIL3(7),INT(100),INTH(200),TITLE(20),O(33)
0007            COMMON /BK0/ NUNK,IHBW,NLINES,NCOLS,NITT,ITTCNT,CONV,
                1RTAS,NAQS,OXFER,IROT,MAXIT,DELX,AUTO,OMEGA,NTMPLN,
                2NOTIMS,TRNOSS,IWID,TIME,STIME,ETIME,YEAR,MONTH,DAY,HOUR,
                3MIN,YEARL,MONTHL,DAYL,HOURL,MINL,NOSX,NOSY,NAOFRS,
                4NUCONF,DOIT,DOIT1,STOMUL,TITL(80)
                5,COFLEK,SEALEV,DELLEK,ITVN,ILRGSM,ISMLSM,
                6XLLH,YLLH,IFAKE(160)
0008            COMMON /RESULT/NUMSAV,RTIMES(5,26),RMIN(26),IFIN1(202)
        C
0009            REAL NEW,MODL,MODU
        C
0010            DATA SKIP /4HNONE/
0011            DATA RNULL /4H    /
0012            DATA BLNK /1H /
0013            DATA INTH1/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
        C
0014            WRITE(6,*) 'PROGRAM DIFER TO MAKE DIFFERENCE MAPS'
0015            WRITE(6,*) 'WHICH DISK UNIT 1,2, 3=DB0:'
0016            READ(6,*) IUN
0017            IF(IUN.NE.3) CALL ASNLUN(1,'DP',IUN,ISTAT)
0018            CALL OPFILE(1,'VTTTEMP',400.,IPTR)
0019    2222    CONTINUE
0020            WRITE(6,*) 'ENTER THE BASE FILE NAME'
0021            WRITE(6,*) 'THEN NEW FILE NAME'
0022            WRITE(6,*) 'THEN DIFFERENCE FILE NAME'
0023            WRITE(6,*) 'THEN MODULO NUMBER,NODES-X,NODES-Y'
0024            READ(6,2,END=18593) FIL1,FIL2,FIL3,MODU,NODESX,NODESY
0025    2       FORMAT(7A4/7A4/7A4/F10.0,2I10)
0026            WRITE(6,3) FIL1,FIL2,FIL3,MODU,NODESX,NODESY
0027    3       FORMAT(1X,7A4/1X,7A4/1X,7A4/F10.0,2I10)
0028            N=ICHR(FIL1)
0029            N=ICHR(FIL2)
0030            N=ICHR(FIL3)
        C CALCULATE THE DIFFERENCE FILE
```

```
0082              IF(INCNT.GT.0) AVGN=AVGN/INCNT
0083              CLOSE(UNIT=5)
0084              CLOSE(UNIT=2)
0085              IF(FIL3(1).NE.'NONE') CLOSE(UNIT=3)
0086              WRITE(6,12) AMAX,AMIN,SQSUM
0087     12       FORMAT(' MAX-MIN-RMS DIFFERENCE '/3E15.5)
0088     1000     CONTINUE
0089              IF(FIL1(1).NE.'NONE') GO TO 98
0090              AMAX=-10000.
0091              AMIN=10000.
0092              AVG=0.0
0093              AVGP=0.0
0094              AVGN=0.0
0095              IPCNT=0
0096              INCNT=0
0097              SQSUM=0.0
0098              ICNT=0
0099              SMALL=0.0001
0100     98       CONTINUE
         C READ IN THE DIFFERENCE FILE
0101              DO 100 I=1,NODESY
0102              ADR=(I-1)*2.
0103              CALL DPR(IPTR,ADR,B,512.)
0104              IF(FIL1(1).NE.'NONE') GO TO 100
0105              DO 99 J=1,NODESX
0106              P=B(J)
0107              IF(P.EQ.0.) GO TO 99
0108              SQSUM=SQSUM+P**2
0109              IF(P.GT.AMAX) AMAX=P
0110              IF(P.LT.AMIN) AMIN=P
0111              AVG=AVG+ABS(P)
0112              ICNT=ICNT+1
0113              IF(P) 95,95,96
0114     96       IPCNT=IPCNT+1
0115              AVGP=AVGP+P
0116              GO TO 99
0117     95       INCNT=INCNT+1
0118              AVGN=AVGN+P
0119     99       CONTINUE
0120     100      CONTINUE
0121              IF(FIL1(1).NE.'NONE') GO TO 94
0122              SQSUM=SQRT(SQSUM/FLOAT(ICNT))
0123              AVG=AVG/ICNT
0124              IF(IPCNT.GT.0) AVGP=AVGP/IPCNT
0125              IF(INCNT.GT.0) AVGN=AVGN/INCNT
0126     94       CONTINUE
         C
0127     3011     CONTINUE
0128              DO 500 I=1,200
0129     500      INTH(I)=I
0130              WRITE(6,*) 'ENTER A TITLE FOR THE PRINTOUT'
0131              READ(6,511) TITLE
0132              WRITE(6,512) (TITLE(III),III=1,18)
0133     511      FORMAT(20A4)
0134              WRITE(6,*) 'ENTER THE NUMBER OF DIFFERENCE MAPS TO BE PREPARED'
0135              READ(5,*) NUMCFY
```

```
0031              IF(FIL1(1).NE.'NONE') GO TO 1008
0032              OPEN(UNIT=3,NAME=FIL3,TYPE='OLD',FORM='UNFORMATTED',READONLY)
0033              DO 1007 IY=1,NODESY
0034              READ(3) (B(I),I=1,NODESX)
0035              ADR=(IY-1)*2.
0036              CALL DPW(IPTR,ADR,B,512.)
0037      1007    CONTINUE
0038              CLOSE(UNIT=3)
0039              GO TO 1000
0040      1008    CONTINUE
0041              OPEN(UNIT=5,NAME=FIL1,TYPE='OLD',FORM='UNFORMATTED'
                  1,READONLY)
0042              OPEN(UNIT=2,NAME=FIL2,TYPE='OLD',FORM='UNFORMATTED'
                  1,READONLY)
0043              IF(FIL3(1).NE.'NONE')
                  1OPEN(UNIT=3,NAME=FIL3,TYPE='NEW',FORM='UNFORMATTED')
          C       IF(FIL3(1).EQ.'NONE')
          C       1OPEN(UNIT=3,NAME='[200,200]TDSS',TYPE='NEW',FORM='UNFORMATTED')
0044              AMAX=-10000.
0045              AMIN=10000.
0046              AVG=0.0
0047              AVGP=0.0
0048              AVGN=0.0
0049              IPCNT=0
0050              INCNT=0
0051              SQSUM=0.0
0052              ICNT=0
0053              SMALL=0.0001
0054              DO 10 I=1,NODESY
0055              READ(5) (BASE(J),J=1,NODESX)
0056              READ(2) (NEW(J),J=1,NODESX)
0057              DO 11 J=1,NODESX
0058              BASE(J)=NEW(J)-BASE(J)+SMALL
0059              IF(ABS(BASE(J)).GT.500.) BASE(J)=SMALL
0060              IF(NEW(J).EQ.0.) BASE(J)=0.0
0061              IF(NEW(J).EQ.0.) GO TO 11
0062              IF(BASE(J).GT.AMAX) AMAX=BASE(J)
0063              IF(BASE(J).LT.AMIN) AMIN=BASE(J)
0064              SQSUM=SQSUM+(BASE(J)-SMALL)**2
0065              AVG=AVG+ABS(BASE(J)-SMALL)
0066              IF(BASE(J)-SMALL) 999,998,1001
0067      1001    IPCNT=IPCNT+1
0068              AVGP=AVGP+(BASE(J)-SMALL)
0069              GO TO 998
0070      999     INCNT=INCNT+1
0071              AVGN=AVGN+(BASE(J)-SMALL)
0072      998     CONTINUE
0073              ICNT=ICNT+1
0074      11      CONTINUE
0075              IF(FIL3(1).NE.'NONE') WRITE(3) (BASE(J),J=1,NODESX)
0076              ADR=(I-1)*2.
0077              CALL DPW(IPTR,ADR,BASE,512.)
0078      10      CONTINUE
0079              SQSUM=SQRT(SQSUM/FLOAT(ICNT))
0080              AVG=AVG/ICNT
0081              IF(IPCNT.GT.0) AVGP=AVGP/IPCNT
```

```
0136              IF(NUMCPY.EQ.0) NUMCPY=1
0137              IF(NUMCPY.GT.4) NUMCPY=1
0138              WRITE(6,*) 'NUMBER OF COPIES=',NUMCPY
0139     512      FORMAT(1X,20A4)
0140              DO 400 ICPY=1,NUMCPY
0141              NPGS=NODESX/25
0142              IF(MOD(NODESX,25).NE.0) NPGS=NPGS+1
0143              DO 400 IX=1,NPGS
0144              ISP=(IX-1)*25+1
0145              ISTP=ISP-1+25
0146              IF(IX.EQ.NPGS.OR.NPGS.EQ.1) ISTP=NODESX
0147              IF(.NOT.(IX.EQ.2.OR.NPGS.EQ.1)) WRITE(4,444)
0148     444      FORMAT(1H1///)
0149              IF(.NOT.(IX.EQ.2.OR.IPGS.EQ.1)) GO TO 1006
0150               WRITE(4,445) TITLE,AMAX,AMIN,SQSUM
0151     445      FORMAT(1H1,7X,20A4/8X,' MAX=',E12.4,' MIN=',E12.4,' RMS=',
                  1E12.5,' PLUS MEANS NEW IS HIGHER')
0152               WRITE(4,446) AVG,ICNT,AVGP,IPCNT,AVGN,INCNT
0153     446      FORMAT(8X,' AVG=',E12.3,
                  1' # ACTIVE NODES=',I5,' AVG POS=',F10.4,' # POS=',I5,
                  2' AVG NEG=',F10.4,' # NEG=',I5)
0154               WRITE(4,4500) FIL1,FIL2
0155     4500     FORMAT(10X,' BASE FILE NAME ' ,7A4,10X,' NEW FILE NAME'
                  1,7A4)
0156     1006     CONTINUE
0157              WRITE(4,450) (INTH(I),I=ISP,ISTP)
0158     450      FORMAT(/' COLUMN  ',100(1H*)/
                  18X,25I4)
0159              WRITE(4,451)
0160     451      FORMAT(' LINE'/)
0161              DO 200 ILIN=1,NODESY
0162              IL=NODESY-ILIN+1
0163              ADR=(IL-1)*2.
0164              CALL DPR(IPTR,ADR,B,512.)
0165              DO 300 J=1,NODESX
0166              SGN=1.0
0167              IF(B(J).NE.0.) SGN=B(J)/ABS(B(J))
0168              INT(J)=B(J)/MODU+.5*SGN
0169              IF(B(J).EQ.0.0) INT(J)=0.0
0170     300      CONTINUE
         C        WRITE(4,350) IL,(INT(J),J=ISP,ISTP)
0171              ENCODE(108,350,O) IL,(INT(J),J=ISP,ISTP)
0172     350      FORMAT(I7,1H*,25I4)
0173              WRITE(4,351)
0174     351      FORMAT(1H )
0175              DO 352 J=1,25
0176              IDX=ISP+J-1
0177              IF(IDX.GT.NODESX) O(J+2)=RNULL
0178              IF(IDX.GT.NODESX) GO TO 352
0179              IF(B(IDX).EQ.0.0) O(J+2)=RNULL
0180     352      CONTINUE
0181              WRITE(4,353) (O(J),J=1,27)
0182     353      FORMAT(27A4)
0183     200      CONTINUE
0184     400      CONTINUE
         C
```

```
        C  PLOTTING RANGES
        C
0185    20112   FORMAT(6(9X,A1)/6F10.2/)
0186            WRITE(6,*) 'ENTER 6 ONE CHARACTER PLOTTING SYMBOLS FOLLOWED BY'
0187            WRITE(6,*) 'SIX UPPER LIMITS ON THE RANGES (IF THE FIRST'
0188            WRITE(6,*) 'PLOTTING CHARACTER IS AN (E) NO SYMBOL DIFFERENCE'
0189            WRITE(6,*) 'MAP WILL BE MADE'
0190            READ(6,2001,END=5583) A,BB,C,D,E,F,P,Q,R,S,T,U
0191    2001    FORMAT(6A1,6F10.0)
0192            IF(A.EQ.'E') GO TO 5583
0193            WRITE(6,20112) A,BB,C,D,E,F,P,Q,R,S,T,U
0194            DO 5584 ICPY=1,NUMCPY
0195            Z=0.0
0196            NUMAP=0
0197            NUMAM=0
0198            NUMBP=0
0199            NUMBM=0
0200            NUMCP=0
0201            NUMCM=0
0202            NUMDP=0
0203            NUMDM=0
0204            NUMEP=0
0205            NUMEM=0
0206            NUMFP=0
0207            NUMFM=0
0208            NPAGE=NODESX/50
0209            IF(MOD(NODESX,50).NE.0) NPAGE=NPAGE+1
0210            DO 2012 IPAGE=1,NPAGE
0211            IF(IPAGE.GT.1) WRITE(4,444)
0212            ISTRT=(IPAGE-1)*50+1
0213            ISTOP=ISTRT+50-1
0214            IF(IPAGE.EQ.NPAGE) ISTOP=NODESX
0215            IF(IPAGE.EQ.1) WRITE(4,445) TITLE,AMAX,AMIN,SQSUM
0216            IF(IPAGE.EQ.1) WRITE(4,446) AVG,ICNT,AVGP,IPCNT,AVGN,INCNT
0217            IF(IPAGE.EQ.1) WRITE(4,4500) FIL1,FIL2
0218            DO 699 IL=1,106
0219            INTH2(IL)=BLNK
0220            INTH3(IL)=BLNK
0221            INTH4(IL)=BLNK
0222    699     CONTINUE
0223            DO 701 IKP=ISTRT,ISTOP
0224            IKX=IKP-ISTRT+1
0225            IKXO=IKX*2+5
0226            IHNS=IKP/100
0227            ITNS=(IKP-IHNS*100)/10
0228            IDGS=(IKP-IHNS*100-ITNS*10)
0229            INTH4(IKXO)=INTH1(IHNS+1)
0230            INTH3(IKXO)=INTH1(ITNS+1)
0231            INTH2(IKXO)=INTH1(IDGS+1)
0232            IF(IHNS.NE.0) GO TO 705
0233            INTH4(IKXO)=BLNK
0234    705     CONTINUE
0235            IF(ITNS.NE.0) GO TO 706
0236            IF(IHNS.EQ.0) INTH3(IKXO)=BLNK
0237    706     CONTINUE
0238    701     CONTINUE
```

```
0239            WRITE(4,703) (INTH4(IJ),IJ=1,106)
0240            WRITE(4,704) (INTH3(IJ),IJ=1,106)
0241            WRITE(4,704) (INTH2(IJ),IJ=1,106)
0242    703     FORMAT(/' COLUMN  ',100(1H*)/4X,106A1)
0243    704     FORMAT(4X,106A1)
0244            WRITE(4,4510)
0245    4510    FORMAT(' LINE',4X,100(1H*))
0246            DO 2011 ILINE=1,NODESY
0247            ILN=NODESY-ILINE+1
0248            ADR=(ILN-1)*2.
0249            CALL DPR(IP1R,ADR,B,512,)
0250            DO 2002 IJ=1,128
0251            OUT(IJ)=BLNK
0252    2002    CONTINUE
0253            ENCODE(8,2003,OUT(1)) ILN
0254    2003    FORMAT(1X,I6,'*')
0255            DO 2009 ICOL=ISTRT,ISTOP
0256            ISPOS=5+(ICOL-ISTRT)*2+1+3
0257            ICPOS=ISPOS+1
0258            IF(B(ICOL).EQ.0.0) GO TO 2009
0259            IF(ABS(B(ICOL)).GT.P) GO TO 2004
0260            OUT(ICPOS)=A
0261            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0262            IF(B(ICOL).LT.0.0) NUMAM=NUMAM+1
0263            IF(B(ICOL).GE.0.0) NUMAP=NUMAP+1
0264            GO TO 2009
0265    2004    IF(ABS(B(ICOL)).GT.Q) GO TO 2005
0266            OUT(ICPOS)=B
0267            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0268            IF(B(ICOL).LT.0.0) NUMBM=NUMBM+1
0269            IF(B(ICOL).GE.0.0) NUMBP=NUMBP+1
0270            GO TO 2009
0271    2005    IF(ABS(B(ICOL)).GT.R) GO TO 2006
0272            OUT(ICPOS)=C
0273            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0274            IF(B(ICOL).LT.0.0) NUMCM=NUMCM+1
0275            IF(B(ICOL).GE.0.0) NUMCP=NUMCP+1
0276            GO TO 2009
0277    2006    IF(ABS(B(ICOL)).GT.S) GO TO 2007
0278            OUT(ICPOS)=D
0279            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0280            IF(B(ICOL).LT.0.0) NUMDM=NUMDM+1
0281            IF(B(ICOL).GE.0.0) NUMDP=NUMDP+1
0282            GO TO 2009
0283    2007    IF(ABS(B(ICOL)).GT.T) GO TO 2008
0284            OUT(ICPOS)=E
0285            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0286            IF(B(ICOL).LT.0.0) NUMEM=NUMEM+1
0287            IF(B(ICOL).GE.0.0) NUMEP=NUMEP+1
0288            GO TO 2009
0289    2008    OUT(ICPOS)=F
0290            IF(B(ICOL).LT.0.0) OUT(ISPOS)='-'
0291            IF(B(ICOL).LT.0.0) NUMFM=NUMFM+1
0292            IF(B(ICOL).GE.0.0) NUMFP=NUMFP+1
0293    2009    CONTINUE
0294            WRITE(4,4510) (OUT(IJ),IJ=1,128)
```

```
0295    2010    FORMAT(1X,128A1)
0296    2011    CONTINUE
0297    2012    CONTINUE
        C
        C  CALCULATE TOTAL POSITIVE AND NEGATIVE RANGES
        C
0298            NATOT=NUMAM+NUMAP
0299            NBTOT=NUMBM+NUMBP
0300            NCTOT=NUMCM+NUMCP
0301            NDTOT=NUMDM+NUMDP
0302            NETOT=NUMEM+NUMEP
0303            NFTOT=NUMFM+NUMFP
0304            NPTOT=NUMAP+NUMBP+NUMCP+NUMDP+NUMEP+NUMFP
0305            NMTOT=NUMAM+NUMBM+NUMCM+NUMDM+NUMEM+NUMFM
0306            NODTOT=NATOT+NBTOT+NCTOT+NDTOT+NETOT+NFTOT
        C
        C  CALCULATE PERCENTAGES OF NODES
        C
0307            PCNTA=FLOAT(NATOT)/FLOAT(NODTOT)*100.0
0308            PCNTB=FLOAT(NBTOT)/FLOAT(NODTOT)*100.00
0309            PCNTC=FLOAT(NCTOT)/FLOAT(NODTOT)*100.00
0310            PCNTD=FLOAT(NDTOT)/FLOAT(NODTOT)*100.00
0311            PCNTE=FLOAT(NETOT)/FLOAT(NODTOT)*100.00
0312            PCNTF=FLOAT(NFTOT)/FLOAT(NODTOT)*100.00
0313            PCNTAP=FLOAT(NUMAP)/FLOAT(NODTOT)*100.00
0314            PCNTBP=FLOAT(NUMBP)/FLOAT(NODTOT)*100.00
0315            PCNTCP=FLOAT(NUMCP)/FLOAT(NODTOT)*100.00
0316            PCNTDP=FLOAT(NUMDP)/FLOAT(NODTOT)*100.00
0317            PCNTEP=FLOAT(NUMEP)/FLOAT(NODTOT)*100.00
0318            PCNTFP=FLOAT(NUMFP)/FLOAT(NODTOT)*100.00
0319            PCNTAM=FLOAT(NUMAM)/FLOAT(NODTOT)*100.00
0320            PCNTBM=FLOAT(NUMBM)/FLOAT(NODTOT)*100.00
0321            PCNTCM=FLOAT(NUMCM)/FLOAT(NODTOT)*100.00
0322            PCNTDM=FLOAT(NUMDM)/FLOAT(NODTOT)*100.00
0323            PCNTEM=FLOAT(NUMEM)/FLOAT(NODTOT)*100.00
0324            PCNTFM=FLOAT(NUMFM)/FLOAT(NODTOT)*100.00
        C
0325            DO 150 I=1,5
0326            BV(I)=0.0
0327            IS(I)=0
0328    150     CONTINUE
0329            BV(5)=0.
        C
        C  FIND 5 LARGEST DIFFERENCES IN DIFFERENCE ARRAY
        C
0330            DO 156 J=1,NODESY
0331            ADR=(J-1)*2.
0332            CALL DPR(IPTR,ADR,B,512.)
0333            DO 156 I=1,NODESX
0334            K=NODESX*(J-1)+I
0335            IF(ABS(B(I)).GE.ABS(BV(5))) GO TO 151
0336            GO TO 156
0337    151     IF(ABS(B(I)).LT.ABS(BV(1))) GO TO 152
0338            BV(5)=BV(4)
0339            BV(4)=BV(3)
0340            BV(3)=BV(2)
```

```
0341              BV(2)=BV(1)
0342              BV(1)=B(I)
0343              IS(5)=IS(4)
0344              IS(4)=IS(3)
0345              IS(3)=IS(2)
0346              IS(2)=IS(1)
0347              IS(1)=K
0348              GO TO 156
0349      152     IF(ABS(B(I)).LT.ABS(BV(2))) GO TO 153
0350              BV(5)=BV(4)
0351              BV(4)=BV(3)
0352              BV(3)=BV(2)
0353              BV(2)=B(I)
0354              IS(5)=IS(4)
0355              IS(4)=IS(3)
0356              IS(3)=IS(2)
0357              IS(2)=K
0358              GO TO 156
0359      153     IF(ABS(B(I)).LT.ABS(BV(3))) GO TO 154
0360              BV(5)=BV(4)
0361              BV(4)=BV(3)
0362              BV(3)=B(I)
0363              IS(5)=IS(4)
0364              IS(4)=IS(3)
0365              IS(3)=K
0366              GO TO 156
0367      154     IF(ABS(B(I)).LT.ABS(BV(4))) GO TO 155
0368              BV(5)=BV(4)
0369              BV(4)=B(I)
0370              IS(5)=IS(4)
0371              IS(4)=K
0372              GO TO 156
0373      155     BV(5)=B(I)
0374              IS(5)=K
0375      156     CONTINUE
          C
0376              WRITE(4,444)
0377              WRITE(4,445) TITLE,AMAX,AMIN,SQSUM
0378              WRITE(4,2030)
0379      2030    FORMAT(//1X,93(1H*)/' RANGE',9X,'VALUE',10X,'ABS DIFF'2X,
                  1'% OF NODES',2X,'POS DIFF',2X,'% OF NODES',2X,'NEG DIFF',2X,
                  2'% OF NODES',/1X,93(1H*))
0380              WRITE(4,2031)A,Z,P,NATOT,PCNTA,NUMAP,PCNTAP,NUMAM,PCNTAM
0381              WRITE(4,2031)BB,P,Q,NBTOT,PCNTB,NUMBP,PCNTBP,NUMBM,PCNTBM
0382              WRITE(4,2031)C,Q,R,NCTOT,PCNTC,NUMCP,PCNTCP,NUMCM,PCNTCM
0383              WRITE(4,2031)D,R,S,NDTOT,PCNTD,NUMDP,PCNTDP,NUMDM,PCNTDM
0384              WRITE(4,2031)E,S,T,NETOT,PCNTE,NUMEP,PCNTEP,NUMEM,PCNTEM
0385              WRITE(4,2031)F,T,U,NFTOT,PCNTF,NUMFP,PCNTFP,NUMFM,PCNTFM
0386      2031    FORMAT(//3X,A1,3X,F10.3,'*',F10.3,4X,I5,5X,F7.3,4X,I5,6X,
                  1F7.3,4X,I5,6X,F7.3)
0387              PCTMTO=PCNTAM+PCNTBM+PCNTCM+PCNTDM+PCNTEM+PCNTFM
0388              PCTPTO=PCNTAP+PCNTBP+PCNTCP+PCNTDP+PCNTEP+PCNTFP
0389              WRITE(4,2032) NOBTOT,NPTOT,PCTPTO,NMTOT,PCTMTO
0390      2032    FORMAT(1X,93(1H*)/' TOTALS',25X,I5,16X,I5,F13.3,4X,I5
                  1,13F13.3)
          C
```

```
0391                WRITE(4,6030)
0392                WRITE(4,603)
0393                IF(FIL1(1).NE.'NONE')
                    1OPEN(UNIT=5,NAME=FIL1,TYPE='OLD',FORM='UNFORMATTED'
                    1,READONLY)
0394                IF(FIL1(1).NE.'NONE')
                    1OPEN(UNIT=2,NAME=FIL2,TYPE='OLD',FORM='UNFORMATTED'
                    1,READONLY)
          C
          C  LOCATE DATA AND MODEL CORRESPONDING TO 5 LARGEST DIFFERENCES
          C
0395                DO 602 N=1,5
0396                I=IS(N)/NODESX+1
0397                M=MOD(IS(N),NODESX)
0398                J=M
0399                IF(I.EQ.1.AND.M.EQ.0) J=IS(N)
0400                IF(M.EQ.0) J=NODESX
0401                BSE=0.
0402                MODL=0.
0403                IF(FIL1(1).EQ.'NONE') GO TO 6011
0404                DO 600 L=1,NODESY
0405                IF(L.NE.I) GO TO 599
0406                READ(5)(BASE(LL),LL=1,NODESX)
0407                READ(2)(NEW(LL),LL=1,NODESX)
0408                BSE=BASE(J)
0409                MODL=NEW(J)
0410                GO TO 601
0411      599       READ(5)
0412                READ(2)
0413      600       CONTINUE
0414      601       REWIND 5
0415                REWIND 2
0416      6011      CONTINUE
0417                ADR=(I-1)*2.
0418                CALL UPR(IPTR,ADR,B,512.)
0419                WRITE(4,604) N,I,J,BSE,MODL,B(J)
0420      602       CONTINUE
0421                IF(FIL1(1).EQ.'NONE') GO TO 5501
0422                CLOSE(UNIT=5)
0423                CLOSE(UNIT=2)
0424      5501      CONTINUE
0425      6030      FORMAT(//33X,20(1H*)/33X,'*FIVE MAXIMUM NODES*',/33X,20(1H*)//)
0426      603       FORMAT(10X,'ROW',10X,'COLUMN',10X,'DATA',15X,'MODEL',14X,
                    1'DIFFERENCE'/5X,82(1H*))
0427      604       FORMAT(5X,I2,3X,I3,11X,I3,7X,E12.5,7X,E12.5,10X,E12.5)
0428                WRITE(4,550)
0429      550       FORMAT(1H1//1H1)
0430      5584      CONTINUE
0431                CLOSE(UNIT=4)
0432                GO TO 2222
0433      5583      CLOSE(UNIT=4)
0434                GO TO 2222
0435      18593     CLOSE(UNIT=4)
0436                STOP
0437                END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

DAYQT

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DB0:DAYGT,LP:/SH=DAYGT,[1,1]BNWLIB/LB
/
ASG=TI:6,LP:5,DB0:1,DP0:2
/
```

```
C ********** DAYQT.FLX *************
C
C PROGRAM TO SET UP A DAILY Q FILE-Q FILE FOR THE VTT TRANSIENT MODEL
C
C DAYQT,LP:/SH=DAYQT,[1,1]BNWLIB/LB
C /
C ASG=TI:6,LP:5,DB0:1,DP0:2
C /
C
      BYTE LRG(30,200),CNT(200),ILRG(30),ISML(30)
C
C COMMON BLOCK FOR THE ADDQS FILE 248. WORDS
      COMMON /ADDQS/IQTIM0(5),NWLLSL,NWLLSS,WLRATL(30),IWLOCL(2,30),
     1WLRATS(30),IWLOCS(2,30),LOST
C
      DIMENSION FILE(7),IAQ(30),XY(2,30),IXYL(2,30),IXYS(2,30),
     1ITLOCL(2,30),ITLOCS(2,30),ILPOS(30),ISPOS(30),RATE(30)
     1,MON(12)
C
      DATA MON/31,28,31,30,31,30,31,31,30,31,30,31/
      WRITE(6,*) 'DAYQT PROGRAM FOR SETTING UP A DAILY PUMPING'
      WRITE(6,*) 'FILE OF ADDITIONAL STRESSES FOR THE TRANSIENT'
      WRITE(6,*) 'VTT PROGRAM'
      WRITE(6,*) 'ENTER THE NAME OF THE INPUT DATA FILE'
      READ(6,1) FILE
1     FORMAT(7A4)
      N=ICHR(FILE)
      OPEN(UNIT=2,NAME=FILE,TYPE='OLD',READONLY)
      WRITE(6,*) 'WHICH FILE-Q DISK 1,2 OR 3=DB0:'
      READ(6,*) IUN
      IF(IUN.NE.3) CALL ASNLUN(1,'OP',IUN,ISTAT)
      CALL OPFILE(1,'VTTADDQS',367.,IADQ)
C READ IN THE DIMENSIONS AND SMALL REGION INFORMATION AND ROTATION
C UNROTATED DIMENSIONS FOR
C NODESX,NODESY,NO. OF AQUIFERS,LOWER LEFT HAND X AND Y COORDS. OF (1,1)
      READ(2,*) NX,NY,NAQS,DELX,XLLH,YLLH
C UNROTATED LINE AND COLUMN NUMBERS
C COL. MIN.,COL. MAX.,LINE MIN.,LINE MAX.,MAGNIFICATION FOR SMALL REGION
      READ(2,*) IX1,IX2,IY1,IY2,MAG
C ROTATION TO BE SETUP 0=NORMAL 1=ROTATED
      READ(2,*) IROT
C STARTING YEAR FOR THESE STRESSES
      READ(2,*) IQTIM0(1)
      IQTIM0(1)=IQTIM0(1)+1900
      IF(MOD(IQTIM0(1),4).EQ.0) MON(2)=29
      IQTIM0(2)=1
      IQTIM0(3)=1
      IQTIM0(4)=12
      IQTIM0(5)=0
C NUMBER OF WELLS (MAX.=30)
      READ(2,*) NWLLS
C LOCATION OF THE WELLS X,Y, AND AQUIFER NUMBER
      READ(2,35) (IAQ(J),(XY(I,J),I=1,2),J=1,NWLLS)
```

```
00054   33    FORMAT(I5,2F10.0)
00055   C NOW CALCULATE THE LINE AND COLUMN LIMITS AND POSITIONS ACCORDING TO
00056   C THE ROTATION SELECTED
00057         XLLHL=XLLH
00058         YLLHL=YLLH
00059         XURHL=XLLH+(NX-1)*DELX
00060         YURHL=YLLH+(NY-1)*DELX
00061         XLLHS=(IX1-1)*DELX+XLLH
00062         XURHS=(IX2-1)*DELX+XLLH
00063         YLLHS=(IY1-1)*DELX+YLLH
00064         YURHS=(IY2-1)*DELX+YLLH
00065         XCL=XLLHL
00066         YCL=YLLHL
00067         XCS=XLLHS
00068         YCS=YLLHS
00069         DXS=DELX/MAG
00070         NXL=NX
00071         NYL=NY
00072         NXS=(IX2-IX1)*MAG+1
00073         NYS=(IY2-IY1)*MAG+1
00074         NWLLSL=0
00075         NWLLSS=0
00076         DO(I=1,NWLLS)
00077          . IF(XY(1,I).GE.XLLHL.AND.XY(1,I).LE.XURHL)
00078          . . IF(XY(2,I).GE.YLLHL.AND.XY(2,I).LE.YURHL)
00079          . . . NWLLSL=NWLLSL+1
00080          . . . IXX=(XY(1,I)-XLLHL+.5*DELX)/DELX+1
00081          . . . IYY=(XY(2,I)-YLLHL+.5*DELX)/DELX+1
00082          . . . IAQADD=NX
00083          . . . IF(IROT.EQ.1)
00084          . . . . ISX=IXX
00085          . . . . ISY=IYY
00086          . . . . IYY=ISX
00087          . . . . IXX=NY+1-ISY
00088          . . . . IAQADD=NY
00089          . . . ...FIN
00090          . . . ITLOCL(1,NWLLSL)=IXX+IAQADD*(IAQ(I)-1)
00091          . . . ITLOCL(2,NWLLSL)=IYY
00092          . . . ILRG(NWLLSL)=I
00093          . . ...FIN
00094          . ...FIN
00095          . IF(XY(1,I).GE.XLLHS.AND.XY(1,I).LE.XURHS)
00096          . . IF(XY(2,I).GE.YLLHS.AND.XY(2,I).LE.YURHS)
00097          . . . NWLLSS=NWLLSS+1
00098          . . . IXX=(XY(1,I)-XLLHS+.5*DXS)/DXS+1
00099          . . . IYY=(XY(2,I)-YLLHS+.5*DXS)/DXS+1
00100          . . . IAQADD=NXS
00101          . . . IF(IROT.EQ.1)
00102          . . . . ISX=IXX
00103          . . . . ISY=IYY
00104          . . . . IYY=ISX
00105          . . . . IXX=NYS+1-ISY
00106          . . . . IAQADD=NYS
00107          . . . ...FIN
00108          . . . ITLOCS(1,NWLLSS)=IXX+IAQADD*(IAQ(I)-1)
00109          . . . ITLOCS(2,NWLLSS)=IYY
```

```
00110          ,  .  .  ISML(NWLLSS)=I
00111          ,  .  ...FIN
00112          ,  ...FIN
00113          ...FIN
00114   D      WRITE(5,*) NWLLSL
00115   D      DO 900 I=1,NWLLSL
00116   D      WRITE(5,*) (ITLOCL(J,I),J=1,2)
00117   D900   CONTINUE
00118   D      WRITE(5,*) NWLLSS
00119   D      DO 800 I=1,NWLLSS
00120   D      WRITE(5,*) (ITLOCS(J,I),J=1,2)
00121   D800   CONTINUE
00122   C NOW SORT THE WELLS BY LINES AND CREATE AN INDEX FILE FOR READING
00123   C IN THE DATA
00124   C ZERO OUT THE WORKING ARRAYS
00125          DO(I=1,200)
00126          ,  CNT(I)=0
00127          ,  DO(J=1,30)
00128          ,  .  LRG(J,I)=0
00129          ,  ...FIN
00130          ...FIN
00131   C NOW SORT OUT THE LARGE REGION WELLS BY LINE
00132          DO(I=1,NWLLSL)
00133          ,  IPLC=ITLOCL(2,I)
00134          ,  CNT(IPLC)=CNT(IPLC)+1
00135          ,  LRG(CNT(IPLC),IPLC)=I
00136          ...FIN
00137   C CNT(I) HAS THE NUMBER OF WELLS ON ANY LARGE LINE
00138   C LRG(J,I) HAS THE LARGE REGION WELL NUMBER
00139   C
00140          K=0
00141          DO(I=1,200)
00142          ,  IF(CNT(I).NE.0)
00143          ,  .  DO(J=1,CNT(I))
00144          ,  .  .  K=K+1
00145          ,  .  .  ILPOS(K)=ILRG(LRG(J,I))
00146          ,  .  .  IWLOCL(1,K)=ITLOCL(1,LRG(J,I))
00147          ,  .  .  IWLOCL(2,K)=ITLOCL(2,LRG(J,I))
00148          ,  .  ...FIN
00149          ,  ...FIN
00150          ...FIN
00151   D      WRITE(5,*) 'LARGE REGION'
00152   D      WRITE(5,*) K
00153   D      DO 1000 I=1,K
00154   D      WRITE(5,*) (IWLOCL(J,I),J=1,2),ILPOS(I)
00155   D1000  CONTINUE
00156   C NOW SORT THE WELLS BY LINES AND CREATE AN INDEX FILE FOR READING
00157   C IN THE DATA
00158   C ZERO OUT THE WORKING ARRAYS
00159          DO(I=1,200)
00160          ,  CNT(I)=0
00161          ,  DO(J=1,30)
00162          ,  .  LRG(J,I)=0
00163          ,  ...FIN
00164          ...FIN
00165   C NOW SORT OUT THE SMALL REGION WELLS BY LINE
```

```
00166          DO(I=1,NWLLSS)
00167          .  IPLC=ITLOCS(2,I)
00168          .  CNT(IPLC)=CNT(IPLC)+1
00169          .  LRG(CNT(IPLC),IPLC)=I
00170          ...FIN
00171    C CNT(I) HAS THE NUMBER OF WELLS ON ANY SMALL LINE
00172    C LRG(J,I) HAS THE SMALL REGION WELL NUMBER
00173    C
00174          K=0
00175          DO(I=1,200)
00176          .  IF(CNT(I).NE.0)
00177          .  .  DO(J=1,CNT(I))
00178          .  .  .  K=K+1
00179          .  .  .  ISPOS(K)=ISML(LRG(J,I))
00180          .  .  .  IWLOCS(1,K)=ITLOCS(1,LRG(J,I))
00181          .  .  .  IWLOCS(2,K)=ITLOCS(2,LRG(J,I))
00182          .  .  ...FIN
00183          .  ...FIN
00184          ...FIN
00185    D     WRITE(5,*) 'SMALL REGION'
00186    D     WRITE(5,*) K
00187    D     DO 2000 I=1,K
00188    D     WRITE(5,*) (IWLOCS(J,I),J=1,2),ISPOS(I)
00189    D2000 CONTINUE
00190    C NOW READ IN THE DAILY PUMPING RATES
00191          NCRDS=NWLLS/8
00192          IF(MOD(NWLLS,8).NE.0) NCRDS=NCRDS+1
00193          NDAYS=0
00194          DO(I=1,12) NDAYS=NDAYS+MON(I)
00195          ITDAY=0
00196          DO(MONTH=1,12)
00197          .  DO(IDAY=1,MON(MONTH))
00198          .  .  ITDAY=ITDAY+1
00199          .  .  READ(2,2) M,ID,IY
00200    D     .  .  WRITE(5,*) M,ID,IY
00201    2     .  .  FORMAT(3I5)
00202          .  .  IF(M.NE.MONTH.OR.ID.NE.IDAY)WRITE(6,*)'TIME ERROR',M,ID,IY,ITDAY
00203          .  .  DO(I=1,NCRDS)
00204          .  .  .  READ(2,3) IC,(RATE(J),J=(I-1)*8+1,MIN(NWLLS,I*8))
00205    3     .  .  .  FORMAT(I1,F9.0,7F10.0)
00206          .  .  .  IF(IC.NE.I) WRITE(6,*) 'INPUT CARD ERROR', M,ID,IY,ITDAY
00207          .  .  ...FIN
00208    C NOW SORT OUT THE RATES AND WRITE THE FILE-Q BLOCK
00209          .  .  DO(I=1,NWLLSL)
00210          .  .  .  WLRATL(I)=RATE(ILPOS(I))*192.5
00211          .  .  ...FIN
00212          .  .  DO(I=1,NWLLSS)
00213          .  .  .  WLRATS(I)=RATE(ISPOS(I))*192.5
00214          .  .  ...FIN
00215    D     .  .  WRITE(5,*) ITDAY
00216    D     .  .  WRITE(5,*) RATE
00217    D     .  .  WRITE(5,*) 'SMALL'
00218    D     .  .  WRITE(5,*) WLRATS
00219    D     .  .  WRITE(5,*) 'LARGE'
00220    D     .  .  WRITE(5,*) WLRATL
00221          .  .  CALL DPW(IADQ,(ITDAY)*1.,IQTIM0,248.)
00222          .  ...FIN
00223          ...FIN
00224          STOP
00225          END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

WIQPG

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DP2:[4,5]WIOPG,LP:/SH=WIOPG,[1,1]BNWLIB/LB
/
ASG=DP0:2
ASG=DB0:1
ASG=TI:6
/
```

```
C ******WIQPG ******
C ****** TASK NAME = WIQPG
C ****** PROGRAM TO READ THE FILE CREATED BY WISAPQ
C ****** FORMAT IT AND PUT IT ON THE FILEQ DISK FOR THE VTT PROGRAM
C ****** AUTHOR A . REISENAUER  BATTELLE NORTHWEST
C  WIQPG,LP:/SH=WIQPG
C ICHR
C /
C ASG=DP0: 2
C ADG=DP1: 1
C ASG=TI: 6
C /
      INTEGER QPTR,QTIME,TIMVAL
      COMMON /TIMR/ NQTS,NTPLNS,TIMVAL(5,25),RATIO(25),NSTEP3(25)
     1,QTIME(5,28),QPTR(28),IFIN(142)
      DIMENSION FILE(7),QS(128),NOTYP(2,128)
      WRITE(6,*)'THE WISAP FILE Q MUST BE SETUP'
      WRITE(6,*)'WHICH FILE Q DISK?  1,2,OR DB0:=3'
      READ(6,*) IUN
      WRITE(6,*)'ENTER NAME OF BINARY Q FILE'
      READ(6,1) FILE
1     FORMAT(7A4)
      IF (IUN.NE.3)CALL ASNLUN(1,'DP',IUN,ISTAT)
      CALL DPFILE(1,'VTTFIELDS',2401.,IPTR)
      CALL DPFILE(1,'VTTQS',11205.,IQPTR)
      CALL DPFILE(1,'VTTTEMP',400.,ITEMP)
      IYR=1978
      WRITE (6,*)' ENTER NODEX,NODEY'
      READ(6,*)NDSX,NDSY
      CALL DPW(IQPTR,11201.,NQTS,512.)
C SET UP THE STEADY STATE Q TIMES
      NQTS=2
      QPTR(1)=1.
      QPTR(2)=2.
      QTIME(1,1)=IYR
      QTIME(1,2)=IYR+10.
      QTIME(2,1)=1
      QTIME(2,2)=1
      QTIME(3,1)=1
      QTIME(3,2)=1
      QTIME(4,1)=1
      QTIME(4,2)=1
      QTIME(5,1)=1
      QTIME(5,2)=1
      RNWDS=NDSX*2.
      CALL DPW(IQPTR,11201.,NQTS,512.)
      N=ICHR(FILE)
      OPEN(UNIT=2,NAME=FILE,TYPE='OLD',FORM='UNFORMATTED',READONLY)
      DO 100 ILN=1,NDSY
      READ(2)(QS(I),I=1,NDSX)
      ADR=ILN
      ADR1=ILN+200.
      ADR2=(ILN-1)*12.+4.
      ADR3=(ILN-1)*12.+12.
      CALL DPR(IPTR,ADR2+1.,NOTYP,RNWDS)
      DO 50 ICL=1,NDSX
```

```
0043            IF(NDTYP(2,ICL).EQ.1)QS(ICL)=0.
0044     50     CONTINUE
0045            CALL DPW(IQPTR,ADR,QS,RNWDS)
0046            CALL DPW(IQPTR,ADR1,QS,RNWDS)
0047            CALL DPW(IPTR,ADR2,QS,RNWDS)
0048            CALL DPW(IPTR,ADR3,QS,RNWDS)
0049     100    CONTINUE
0050            CLOSE(UNIT=2)
0051            STOP
0052            END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

WISAPQ

THIS PAGE INTENTIONALLY
LEFT BLANK

```
DP1:[4,5]WISAPQ,LP:/SH=WISAPQ,[1,1]BNWLIB/LB
/
ASG=DB0:2
ASG=DP0:3
ASG=TI:6
/
```

```
C********** WISAPQ **********
C        CREATED AUG 1,78
C       GENERATES A  BINARY FILE CONTAINING FLUXES
C       OUTPUT IS IN CFD READY TO BE READ ONTO THE BASIC DATA DISK FOR VTT
C       ************AUTHOR A. E. REISENAUER**************
        BYTE STAT
        COMMON /BLK/TRNSL(128),IOUT(256),VAL(256)
        DIMENSION QFLUX(128),A(6000),NOTYP(2,128),ARG(4),FILE(7),FLE2(7)
        DATA ARG/1.,0.5,0.375,0.625/
        WRITE(6,*)'WHICH FILE Q DISK?   1,2 OR D80:=3'
        READ(6,*)IUN
        WRITE (6,*)'ENTER CARD FILE NAME'
        READ(6,1)FILE
        WRITE(6,*)' WRITE OUTPUT FLUX FILE NAME'
    1   FORMAT(7A4)
        READ(6,1)FLE2
        N=ICHR(FILE)
        NN=ICHR(FLE2)
        OPEN(UNIT=4,NAME=FILE,TYPE='OLD',FORM='FORMATTED',READONLY)
        OPEN(UNIT=3,NAME=FLE2,TYPE='NEW',FORM='UNFORMATTED')
        IF(IUN.EQ.0)IUN=1
        IF(IUN.NE.3)CALL ASNLUN(2,'DP',IUN)
        CALL DPFILE (2,'VTTFIELDS',2401.,IPT4)
        CALL DPFILE(2,'VTTEMP1',400.,IPT)
        CALL DPFILE(2,'VTTEMP2',400.,IPTR3)
        WRITE(6,*)'NODESX=,NODESY='
        READ(6,*)NODX,NODY
        WRITE(6,*)' DO YOU WISH TO ROTATE THE MATRIX. 1=YES'
        READ(6,*) IRTAT
        NODESY=NODY
        NY=NODY
        WRITE(6,*)' TYPE IN A CFACT FOR CONVERSION OF INPUT TO CFD UNITS'
        READ (6,*)CFACT
        NODESX=NODX
        NADS=1
        IAD=1
        WRITE(6,*)NODESY,NODESX
C  INPUT DISPOSAL FROM COLUMN OF CHANGE ROUTINE
        IDDD=1
  101   READ(4,51) IB,IT,KKK
   51   FORMAT(3I3)
        WRITE (6,51)IB,IT,KKK
        READ(4,36) (IOUT(I),TRNSL(I),I=1,KKK)
   36   FORMAT(5(I5,F10.0))
        DO 360 ICHNG=1,KKK
        IS1=IOUT(ICHNG)
        IS2=IOUT(ICHNG+1)
        IF((ICHNG+1).GT.KKK) IS2=NODESX
        VAL1=TRNSL(ICHNG)
        DO 360 ICOL=IS1,IS2
        VAL(ICOL)=VAL1* CFACT
  360   CONTINUE
        DO 361 ILIN=IB,IT
        ADR=(ILIN-1)*2.
        IF(IRTAT.NE.1)CALL DPW(IPT,ADR,VAL,NODX*2.)
C        WRITE WATER VOLUME TO FILQ DISK TEMPORARY LOCATION
```

```
0050              IF(IRTAT.EQ.1)CALL DPW(IPTR3,ADR,VAL,NODX*2.)
0051      361     CONTINUE
0052      362     CONTINUE
0053              IF(IT.LT.NODESY) GO TO 101
0054              IF(IRTAT.NE.1)GO TO 48
0055      30      NLIN=6000/NODESX
0056              IF(NLIN.GT.NODESY)NLIN=NODESY
0057              NPGS=NODESY/NLIN
0058              IF(MOD(NODESY,NLIN).NE.0)NPGS=NPGS+1
0059              DO 4 IPG=1,NPGS
0060              ISTLN=(IPG-1)*NLIN+1
0061              ISTPLN=ISTLN-1+NLIN
0062              IF(ISTPLN.GT.NODESY)ISTPLN=NODESY
0063              DO 44 ILIN=1,ISTPLN-ISTLN+1
0064              ADR=(ISTLN+ILIN-2)*2.
0065              IDX1=(ILIN-1)*NODESX+1
0066              IDX2=ILIN*NODESX
0067              CALL DPR(IPTR3,ADR,A(IDX1),NODESX*2.)
0068      44      CONTINUE
          C   ROTATE THE Q-MATRIX ON THE DISK ALL OTHER Q WILL COME IN ROTATED
          D       WRITE(6,*)NODESY
0069              WORDS=(NODESY*2.)*NAQS
0070              DO 45 ILIN=1,NODESX
0071              ADR=(ILIN-1)*2.
0072              CALL DPR(IPT,ADR,VAL,WORDS)
0073              DO 46 ICOL=ISTPLN,ISTLN,-1
          C       K=(ICOL-1)*NODESX+ILIN
0074              K=(ICOL-ISTLN)*NODESX+ILIN
          C       J=(IAQ-1)*NODESY+ICOL
0075              J=(IAQ-1)*NODESY+NODESY-ICOL+1
0076              VAL(J)=A(K)
0077      46      CONTINUE
0078              CALL DPW(IPT,ADR,VAL,WORDS)
0079      45      CONTINUE
0080      4       CONTINUE
0081      48      NYROT=NODY
0082              NXROT=NODX
0083              IF(IRTAT.EQ.1)NXROT=NODY
0084              IF(IRTAT.EQ.1)NYROT=NODX
0085              DO 8 J=1,NYROT
0086              ADR=(J-1)*2
0087              CALL DPR(IPT,ADR,QFLUX,256.)
0088              WRITE(3)(QFLUX(I),I=1,NXROT)
0089      8       CONTINUE
0090              CLOSE(UNIT=1)
0091              CLOSE(UNIT=2)
0092              CLOSE (UNIT=4)
0093              CLOSE (UNIT=3)
0094              STOP
0095              END
```

THIS PAGE INTENTIONALLY
LEFT BLANK

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
A. E. Reisenauer (10)
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)

THIS PAGE INTENTIONALLY
LEFT BLANK