```
C ********************************************
C                                            *
C          TM3    SUBROUTINES                *
C                                            *
C ********************************************
C& I2CH
      SUBROUTINE I2CH(I6,C6)
      CHARACTER*6 C6
      CHARACTER*7 C7

      I7=1000000+I6
      WRITE(C7,100)I7
      C6=C7(2:7)
      RETURN
100   FORMAT(I7)
      END
C& SUMAR
      SUBROUTINE SUMAR(A,M,SA)
      DIMENSION A(M)
      S1=0.
      DO J=1,M
      S1=S1+A(J)
      ENDDO
      SA=S1
      RETURN
      END
C& PASINT
      SUBROUTINE PASINT(ISTEP,B,M,LB,A1,A2,LA)
      DIMENSION B(M,LB),A1(M,LA),A2(M,LA)
C
C              IB=LB-LA
C                |
C     1          |             LB
C     xxxxxxxxxxoooooooooooooo      B(M,LB)
C               oooooooooooooo    A1(M,LA)
C               oooooooooooooo    A2(M,LA)
C               1             LA
C
C
C        A1       A2      B
C         x-------o-------x
C

      IB=LB-LA

      IF(MOD(ISTEP,2).NE.0)THEN
           DO K=1,LA
           DO J=1,M
           A2(J,K)=0.5*(A1(J,K)+B(J,IB+K))
           ENDDO
           ENDDO
      ELSE
           DO K=1,LA
           DO J=1,M
           A2(J,K)=B(J,IB+K)
           ENDDO
           ENDDO
      ENDIF
      RETURN
      END
C& FLXCO2
      SUBROUTINE FLXCO2(ISTEP,QFLUX,QFLX,EFIREA,KFIRE)
C
C     Modified 15/01/99
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (JAM=6+2*(JM-10))
```



```
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KKMAX=LM)
      PARAMETER (KMAX=KKMAX)
      PARAMETER (KMIN = LM-KKMAX)


      DIMENSION QFLUX(IMJM,2),EFIREA(IMJM,4)
C-----------------------------------------------------------------------
                                   C O M M O N /LOOPS/
     & KHLA  (JAM),KHHA  (JAM),KVLA  (JAM),KVHA  (JAM)
C
     &,KHL2  (JM),KHH2  (JM),KVL2  (JM),KVH2  (JM)
C
     &,LMH   (IMJM),LMV   (IMJM)
C-----------------------------------------------------------------------
                                    C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------------------
                                   C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,U(IMJM,LM),V(IMJM,LM),T(IMJM,LM),Q(IMJM,LM)
C-----------------------------------------------------------------------
      COMMON/ETATRN/
     * QQL(IMJM,KKMAX),QQH(IMJM,KKMAX),
     * PS(IMJM,2), QQ(IMJM,KKMAX,2),
     * UU(IMJM,KKMAX,2), VV(IMJM,KKMAX,2),
     * WW(IMJM,KKMAX,2), RO(IMJM,KKMAX,2),
     * TT(IMJM,KKMAX,2), ZZ(IMJM,KKMAX+1,2),
     * ALFA(IMJM,KKMAX,2),
     * DIFQ(IMJM,KKMAX),
     * AKH(IMJM,KKMAX,2),AKHS(IMJM,2),
     * TOPG(IMJM)
     *,PREC(IMJM,2),ACPR(IMJM,2)
C-----------------------------------------------------------------------
      DATA RGAS/287.04/
C-----------------------------------------------------------------------


      IHR=MOD(ISTEP,30)

      IF(IHR.EQ.1
     *.OR.
     * IHR.EQ.11
     *.OR.IHR.EQ.15
     *.OR.IHR.EQ.18
     *.OR.IHR.EQ.21)THEN
      KFIRE=4
      IF(IHR.GE.11)KFIRE=1
      IF(IHR.GE.15)KFIRE=2
      IF(IHR.GE.18)KFIRE=3
      IF(IHR.GE.21)KFIRE=4
      CALL CO2F(QFLUX,QFLX,EFIREA(1,KFIRE))
C
C                 QFLUX(*,2)
C
       DO10 J=1,IMJM

       C=PD(J)*RES(J)
       K=LMH(J)
       IF(K.EQ.0)THEN
       QFLUX(J,2)=0.
       QFLUX(J,1)=0.
```

```
       ELSE
       PSUR=PT+AETA(K+KMIN)*C
       TEM=TT(J,K,2)
       ROP=PSUR/(RGAS*TEM)
       DZ=ZZ(J,K,2)-ZZ(J,K+1,2)
       QFLUX(J,2)=QFLUX(J,1)/(ROP*DZ)
       ENDIF

 10    CONTINUE
C
      WRITE(66,*)' ***********************************'
      WRITE(66,*)'ISTEP=',ISTEP,' /* KFIRE=',KFIRE,' */'
      WRITE(66,*)' ***********************************'
      ENDIF

      RETURN
      END
C& FLUXSR
      SUBROUTINE FLUXSR(NU,ISTEP,QFLUX,QFLX,EFIREA,BUFI,IRECF)
C
C     Modified 03/05/99      From FLXCO2
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (JAM=6+2*(JM-10))
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KKMAX=LM)
      PARAMETER (KMAX=KKMAX)
      PARAMETER (KMIN = LM-KKMAX)


      DIMENSION QFLUX(IMJM,2),EFIREA(IMJM,4),BUFI(IMJM,2)
C-----------------------------------------------------------------------
                               C O M M O N /LOOPS/
     & KHLA  (JAM),KHHA  (JAM),KVLA  (JAM),KVHA  (JAM)
C
     &,KHL2  (JM),KHH2  (JM),KVL2  (JM),KVH2  (JM)
C
     &,LMH   (IMJM),LMV   (IMJM)
C------------------------------------------------------------------------
                                C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------------------
                               C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,U(IMJM,LM),V(IMJM,LM),T(IMJM,LM),Q(IMJM,LM)
C-----------------------------------------------------------------------
      COMMON/ETATRN/
     * QQL(IMJM,KKMAX),QQH(IMJM,KKMAX),
     * PS(IMJM,2), QQ(IMJM,KKMAX,2),
     * UU(IMJM,KKMAX,2), VV(IMJM,KKMAX,2),
     * WW(IMJM,KKMAX,2), RO(IMJM,KKMAX,2),
     * TT(IMJM,KKMAX,2), ZZ(IMJM,KKMAX+1,2),
     * ALFA(IMJM,KKMAX,2),
     * DIFQ(IMJM,KKMAX),
     * AKH(IMJM,KKMAX,2),AKHS(IMJM,2),
     * TOPG(IMJM)
     *,PREC(IMJM,2),ACPR(IMJM,2)
C-----------------------------------------------------------------------
C
```



```
      COMMON/ETAREA/CAREA(IMJM)
C
      DATA RGAS/287.04/
C-----------------------------------------------------------------------

      IHR=MOD(ISTEP,30)

C      CALL CO2F(QFLUX,QFLX,EFIREA(1,KFIRE))

      CALL RDFRS(NU,ISTEP,IRECF,BUFI,EFIREA,IMJM)
C
C                 QFLUX(*,1)
C
C
C     EFIREA(J) in km**2
C     CAREA(J)  in m**2
C
      C1=QFLX*1.E6  ! For converting EFIREA(J) to m**2
      C2=C1*1.E6    ! For converting to ppm
      DO J=1,IMJM
      QFLUX(J,1)=C2*EFIREA(J,1)/CAREA(J)
      EFIREA(J,2)=0.
      EFIREA(J,3)=0.
      EFIREA(J,4)=0.
      ENDDO
C
C                 QFLUX(*,2)
C
       DO10 J=1,IMJM

       C=PD(J)*RES(J)
       K=LMH(J)
       IF(K.EQ.0)THEN
       QFLUX(J,2)=0.
       QFLUX(J,1)=0.
       ELSE
       PSUR=PT+AETA(K+KMIN)*C
       TEM=TT(J,K,2)
       ROP=PSUR/(RGAS*TEM)
       DZ=ZZ(J,K,2)-ZZ(J,K+1,2)
       QFLUX(J,2)=QFLUX(J,1)/(ROP*DZ)
       ENDIF

 10    CONTINUE
C
      RETURN
      END
C& WETSED
      SUBROUTINE WETSED(Q,ACPR,DQDT,HTM,KMAX,DTT)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1, LM1=LM-1)
      DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),
     *          ACPR(IMJM,2),DQDT(IMJM,KMAX)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
C      DATA S/0.00046/
      DATA S/0.0046/
C     ----------------------------------------
C     Wet sedimentation rate R= S*(dP/dt)
C     S = 1 cm/hour
C     (dP/dt) in m/(6 hours)
C     S = 100 /(6 * 3600) S   = 0.0046 S cm/s
C     ----------------------------------------
      S1=S*DTT       !    *DTT
      DO5100 KK=1,IPN
      K=INP(KK)

      DO L=1,KMAX
```

```
      DQ=S1*(ACPR(K,2)-ACPR(K,1))*HTM(K,L)
      QP=HTM(K,L)*Q(K,L)
      QNEW=QP-DQ
      QQ=AMAX1(QNEW,0.)
      DQ1=QP-QQ
      DQDT(K,L)=HTM(K,L)*DQ1
      Q(K,L)=HTM(K,L)*QQ

      ENDDO

5100  CONTINUE
      RETURN
      END
C& PROGON
      SUBROUTINE PROGON(A,B,C,F,M,X,BET,Z)
C ------------------------------------------------
C
C     -A(I-1)*X(I-1)+B(I)*X(I)-C(I+1)*X(I+1)=F(I)
C                  I = 1, ..., M
C      A(1)=C(M)=0.,  B(I) >= A(I) + C(I)
C
C ------------------------------------------------
      DIMENSION A(M),B(M),C(M),X(M),F(M),
     * BET(M+1),Z(M+1)
C     ------------
C     FORWARD RUN
C     ------------
      BET(1)=0.
      Z(1)=0.
      DO I=1,M
      BET(I+1)=C(I)/(B(I)-A(I)*BET(I))
      Z(I+1)=(A(I)*Z(I)+F(I))/(B(I)-A(I)*BET(I))
      ENDDO
C
C         WRITE(66,*)'FORWARD RUN IS DONE'
C     ------------
C     BACKWARD RUN
C     ------------
      X(M)=Z(M+1)
      DO I=M-1,1,-1
      X(I)=BET(I+1)*X(I+1)+Z(I+1)
      ENDDO
      RETURN
      END
C& VSDIF2
      SUBROUTINE VSDIF2(Q,VDQDT,HTM,LMH,AKH,ZINT,KMAX,QFLUX,DTQ2)
C     ****************************************************************
C     *                                                              *
C     *  ROUTINE FOR VERTICAL TURBULENT MIXING OF LARGE SCALE        *
C     *  HISTORICAL VARIABLES                                        *
C     *                                                              *
C     *  SEE THE COMMENTS IN PROFQ2                                  *
C     *                                                              *
C     *  Modify at 26/08/98 for TRNETA                               *
C     ****************************************************************
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1, LM1=LM-1)
      DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),LMH(IMJM),
     *AKH(IMJM,KMAX),ZINT(IMJM,KMAX+1),QFLUX(IMJM),
     *VDQDT(IMJM,KMAX)
C-----------------------------------------------------------------------
                               D I M E N S I O N
     &  CL(2:LM),CM(LM),CU(LM1),RV(LM)
     &,  SOLV(LM),AU(LM)
C-----------------------------------------------------------------------
C23456789 123456789 123456789 123456789 123456789 123456789 123456789 12
C        10        20        30        40        50        60        70
C-----------------------------------------------------------------------
```



```
C                             C O M M O N
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
C-----------------------------------------------------------------------
      WRITE(66,*)'VSDIF2 START'
      NCOL=1
      DO5100 KK=1,IPN
      K=INP(KK)
C-----------------------------------------------------------------------
C***
C***  WE BEGIN BY COMPUTING
C***  THE COEFFICIENTS
C***  OF THE TRIDIAGONAL MATRIX
C***  CO2 DIFFUSION.
C***
C
C.....The loops below, up to 'ADD EFFECT OF SURFACE FLUX...' have been
C.....slightly re-arranged from the 'natural' order to take advantage
C.....of multi-tasking.
C
C***                    CL LOWER DIAGONAL
C***
C***                    CU UPPER DIAGONAL
C***
C***                    CM MAIN  DIAGONAL
C***
      DO 1000 L=2,KMAX-1       !      LM1
      CL(L)=HTM(K,L)*(-2.*AKH(K,L-1)*DTQ2)/
     1         ((ZINT(K,L)-ZINT(K,L+1))*(ZINT(K,L-1)-ZINT(K,L+1)))
      CU(L)=HTM(K,L+1)*(-2.*AKH(K,L)*DTQ2)/
     1         ((ZINT(K,L)-ZINT(K,L+1))*(ZINT(K,L)-ZINT(K,L+2)))
      CM(L)=1.+(2.*DTQ2/(ZINT(K,L)-ZINT(K,L+1)))*
     1         (HTM(K,L)*AKH(K,L-1)/(ZINT(K,L-1)-ZINT(K,L+1))
     2         +HTM(K,L+1)*AKH(K,L)/(ZINT(K,L)-ZINT(K,L+2)))
 1000 CONTINUE
C
C
C                    End cases; L=LM, L=1.
C
      L=KMAX         !   LM
      K=INP(KK)
      CL(L)=HTM(K,L)*(-2.*AKH(K,L-1)*DTQ2)/
     1         ((ZINT(K,L)-ZINT(K,L+1))*(ZINT(K,L-1)-ZINT(K,L+1)))
      CU(1)=HTM(K,1+1)*(-2.*AKH(K,1)*DTQ2)/
     1         ((ZINT(K,1)-ZINT(K,1+1))*(ZINT(K,1)-ZINT(K,1+2)))
      CM(1)=1.+(2.*AKH(K,1)*DTQ2)/
     1         ((ZINT(K,1)-ZINT(K,1+1))*(ZINT(K,1)-ZINT(K,1+2)))
      CM(L)=1.+(2.*DTQ2/(ZINT(K,L)-ZINT(K,L+1)))*
     1         (HTM(K,L)*AKH(K,L-1)/(ZINT(K,L-1)-ZINT(K,L+1)))
C***
C***  NOW SOLVE FOR DIFFUSION OF WATER VAPOR
C***
C***   SET UP RIGHT HAND SIDE VECTOR FOR SPECIFIC HUMIDITY
C***
      DO 1090 L=1,KMAX      !    LM
      RV(L)=Q(K,L)
 1090 CONTINUE
      L=LMH(K)
      RV(L)=Q(K,L)+DTQ2*QFLUX(K)
C***
C***                    LMH LAST LAYER ABOVE GROUND
C***  MODIFY ONLY CM(K,LMH) TO REPLACE HEAT FLUX WITH VAPOR FLUX
C***
C***      L=LMH(K)
C***      RV(L)=RV(L)+2.*DTQ2*CKLQ(K)*QZ0(K)/
C***     1                    (ZINT(K,L)-ZINT(K,L+1))
C***      CM(L)=CM(L)+2.*DTQ2*(CKLQ(K)-FFS(K))/
C***     1          (ZINT(K,L)-ZINT(K,L+1))
C***
C***        NOW CALL SOLUTION OF DIFFUSION EQUATION
```

```
C***

      CALL TRIDI(NCOL,1,1,KMAX,CL,CM,CU,RV,SOLV,AU)
C                          LM

C***
C***  SAVE THE SOLUTION IN Q(K,L) CO2 SOLUTION
C***
      DO 1110 L=1,KMAX                  !  LM
      VDQDT(K,L)=HTM(K,L)*(SOLV(L)-Q(K,L))
      Q(K,L)=SOLV(L)
 1110 CONTINUE
 5100 CONTINUE
       WRITE(66,*)'VSDIF2 FINISHED'
C
      RETURN
      END
C& CHKPT
      SUBROUTINE CHKPT(NU,WR,IDAT,ISTEP,IREC,IREC1,IREC2,QQ,
     *                                              IMJM,KMAX)
      DIMENSION QQ(IMJM,KMAX),  IDAT(3)
      CHARACTER*1 WR
      REWIND(NU)
      IF(WR.EQ.'R')THEN
      READ(NU)IDAT,ISTEP,IREC,IREC1,IREC2,
     *                ((QQ(I,J),I=1,IMJM),J=1,KMAX)
      RETURN
      ENDIF
      IF(WR.EQ.'W')THEN
      WRITE(NU)IDAT,ISTEP,IREC,IREC1,IREC2,
     * ((QQ(I,J),I=1,IMJM),J=1,KMAX)
      CALL ADDCHP(NU)
      RETURN
      ENDIF
      WRITE(66,*)'ERROR IN CHKPT:',WR
      STOP
      END
CX C& ADDCHP
CX       SUBROUTINE ADDCHP(NU)
CX       PARAMETER (IM=111,JM=221,LM=38,LSM=19)
CX       PARAMETER (IMJM = IM*JM - JM/2)
CX       PARAMETER (LP1=LM+1)
CX       PARAMETER (JAM=6+2*(JM-10))
CX       PARAMETER (IMAX=2*IM-1  , JMAX=JM)
CX       PARAMETER (KKMAX=LM)
CX       PARAMETER (KMAX=KKMAX)
CX       PARAMETER (KMIN = LM-KKMAX)
CX       PARAMETER (KTOP = KMIN + 1)
CX C-----------------------------------------------------------------------
CX                                  C O M M O N /VRBLS/
CX      & PD(IMJM),RES(IMJM),FIS(IMJM)
CX C
CX      &,U(IMJM,LM),V(IMJM,LM),T(IMJM,LM),Q(IMJM,LM)
CX C-----------------------------------------------------------------------
CX                                   C O M M O N /MAPOT/
CX      & LSL,IXM,IYM
CX      &,ISHDE(49)
CX      &,R,PT,TSPH
CX      &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
CX      &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
CX      &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
CX      &,ERLAM0,CPHI0,SPHI0
CX      &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
CX C-----------------------------------------------------------------------
CX       DIMENSION WA(IMJM)
CX C
CX       DO J=1,IMJM
CX       WA(J)=PD(J)*RES(J)
CX       ENDDO
```



```
CX       WRITE(NU)WA
CX       WRITE(NU)DETA
CX       RETURN
CX       END
C& ADDCHP
      SUBROUTINE ADDCHP(NU)
C
C     Aerosol mass calculation for an atmospheric
C     column with an unit area and writting to CHKPT
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (JAM=6+2*(JM-10))
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KKMAX=LM)
      PARAMETER (KMAX=KKMAX)
      PARAMETER (KMIN = LM-KKMAX)
      PARAMETER (KTOP = KMIN + 1)
C-----------------------------------------------------------------------
                             C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,U(IMJM,LM),V(IMJM,LM),T(IMJM,LM),Q(IMJM,LM)
C-----------------------------------------------------------------------
                              C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------------------
                             C O M M O N /MASKS/
     & HBM2  (IMJM),VBM2  (IMJM),VBM3  (IMJM),SM    (IMJM),SICE (IMJM)
C
     &,HTM   (IMJM,LM),VTM   (IMJM,LM)
C ----------------------------------------------------------------------
      COMMON/ETATRN/
     * QQL(IMJM,KKMAX),QQH(IMJM,KKMAX),
     * PS(IMJM,2), QQ(IMJM,KKMAX,2),
     * UU(IMJM,KKMAX,2), VV(IMJM,KKMAX,2),
     * WW(IMJM,KKMAX,2), RO(IMJM,KKMAX,2),
     * TT(IMJM,KKMAX,2), ZZ(IMJM,KKMAX+1,2),
     * ALFA(IMJM,KKMAX,2),
     * DIFQ(IMJM,KKMAX),
     * AKH(IMJM,KKMAX,2),AKHS(IMJM,2),
     * TOPG(IMJM)
     *,PREC(IMJM,2),ACPR(IMJM,2)
C ----------------------------------------------------------------------
      DIMENSION WA(IMJM)
C ----------------------------------------------------------------------
C
      DATA G /9.81/
C
       DO J=1,IMJM
       WA(J)=(PS(J,1)-PT)*RES(J)
       ENDDO
       WRITE(NU)WA
       WRITE(NU)DETA

      DO J=1,IMJM
      PPS=(PS(J,1)-PT)*RES(J)
      C1=1.E-3*PPS/G
      S=0.

      DO K=1,KKMAX
```

```
      S=S+C1*QQ(J,K,1)*DETA(K)*HTM(J,K)
      ENDDO

      WA(J)=S

      ENDDO
      WRITE(NU)WA
      RETURN
      END
C& EPRESS
      SUBROUTINE EPRESS(PETAL,RES,PD,AETA,PT)
      PARAMETER (IM=111,JM=221,LM=38)
      PARAMETER (IMJM = IM*JM - JM/2)

      DIMENSION PETAL(IMJM,LM),RES(IMJM),PD(IMJM),AETA(LM)
C
C     PRESSURE ON ETA LEVELS
C
       DO10 J=1,IMJM
       C=PD(J)*RES(J)

       DO20 K=1,LM
       PETAL(J,K)=PT+AETA(K)*C
 20    CONTINUE

 10    CONTINUE
      RETURN
      END
C& CORW
      SUBROUTINE CORW(WETA,IMJM,LM)
      DIMENSION WETA(IMJM,LM)

      DO J=1,IMJM
      WETA(J,LM)=WETA(J,LM-1)
      ENDDO

      DO K=LM-1,2,-1
      DO J=1,IMJM
      WETA(J,K)=0.5*(WETA(J,K)+WETA(J,K-1))
      ENDDO
      ENDDO

      DO J=1,IMJM
      WETA(J,1)=0.5*WETA(J,1)
      WETA(J,LM)=0.5*WETA(J,LM)
      ENDDO

      RETURN
      END
C& DIAGWW
      SUBROUTINE DIAGWW(WETA,HTM,IMJM,LM)
      DIMENSION WETA(IMJM,LM), HTM(IMJM,LM)
      LOGICAL LSURF
      S=0.
      N=0
      SMN=-1.E6
      SMN= 1.E6
      LM1=LM-1
      DO J=1,IMJM
      DO K=1,LM1
      LSURF=HTM(J,K).EQ.1.0
     *       .AND.
     *       HTM(J,K+1).EQ.0.0
      IF(LSURF)THEN
      N=N+1
      R=WETA(J,K)
      S=S+R
      IF(R.GT.SMX)SMX=R
      IF(R.LT.SMN)SMN=R
```



```
      ENDIF
      ENDDO
      ENDDO
      IF(N.NE.0)THEN
      S=S/FLOAT(N)
      WRITE(66,*)'ETADOT ON SURFACE:',S,SMN,SMX,N
      ELSE
      WRITE(66,*)'ETADOT ON SURFACE:',S,SMN,SMX,N
      ENDIF
      RETURN
      END
C& ALFMAX
      SUBROUTINE ALFMAX(QH,QL,QM,QSUR,NQSUR,ALF)
C
C     Routine calculates QM and ALF for one point
C     Eta version
C
C     Case NQSUR = 0?
C
      DIMENSION QSUR(8)
C
      DU=QH-QL                              ! UH-UL
      IF(DU.EQ.0.)THEN
      AL=1.
      QM=QL
      ELSE

      QMX=QSUR(1)
      QMN=QSUR(1)

          IF(NQSUR.EQ.0)THEN
CX          WRITE(66,*)'ALFMAX ERROR?: QH=',
CX     *            QH,' QL=',QL,' NQSUR=',NQSUR
C23456
          AL=1.
          QM=AL*QH+(1.-AL)*QL
          ALF=AL
          RETURN
C          STOP 'ERROR IN ALFMAX'
          ENDIF

      DO J=1,NQSUR
      IF(QSUR(J).GT.QMX)QMX=QSUR(J)
      IF(QSUR(J).LT.QMN)QMN=QSUR(J)
      ENDDO

      ALMIN=(QMN-QL)/DU
      ALMAX=(QMX-QL)/DU

      AL=AMAX1(ALMIN,ALMAX)
      IF(AL.GT.1.)AL=1.
      IF(AL.LT.0.)AL=0.

      QM=AL*QH+(1.-AL)*QL
      ALF=AL
      ENDIF

      RETURN
      END
C& CDIST
      REAL FUNCTION CDIST(X1,X2)
C
C     Euclidean distance between two points on line
C
      CDIST=ABS(X1-X2)
      RETURN
      END
C& DIST
      REAL FUNCTION DIST(X1,Y1,X2,Y2)
```

```
C
C     Euclidean distance between two points
C
      R=(X1-X2)**2+(Y1-Y2)**2
      DIST=SQRT(R)
      RETURN
      END
C& INDEXS
      SUBROUTINE INDEXS(K,IMAX,IM,I,J)
C     I, J INDEX FROM K ADDRESS
      KMOD=MOD(K,IMAX)
      KINT=K/IMAX
      IF(KMOD.EQ.0)THEN
      I=IMAX-1
      J=2*KINT
      ELSE
          IF(KMOD.LE.IM)THEN
          I=2*KMOD-1
          J=2*KINT+1
          ELSE
          I=2*KMOD-2*IM
          J=2*KINT+2
          ENDIF
      ENDIF
      RETURN
      END
C& GETMSK
      SUBROUTINE GETMSK(HM,IMJM,IM,I1,I2,J1,J2)
      DIMENSION HM(IMJM)
      IMAX=2*IM-1
      DO K=1,IMJM
      CALL INDEXS(K,IMAX,IM,I,J)
      IF(I.GE.I1.AND.I.LE.I2
     *          .AND.
     *   J.GE.J1.AND.J.LE.J2)THEN
      HM(K)=1.
      ELSE
      HM(K)=0.
      ENDIF
      ENDDO
      RETURN
      END
C& REAVAR
      SUBROUTINE REAVAR(NUWRI,NAME10,VAR,IMJM,LM)
      CHARACTER*10 NAME10
      DIMENSION VAR(IMJM,LM)
      DO K=1,LM
      READ(NUWRI)NAME10,IMJM1,K1,LM1,(VAR(I,K),I=1,IMJM)
      ENDDO
      RETURN
      END
C& EXTR2M
      SUBROUTINE EXTR2M(NU,NAME,HA,HM,HG,IMJM,KMAX,ICL)
      DIMENSION HA(IMJM,KMAX),HM(IMJM,KMAX),HG(IMJM)
      CHARACTER*(*) NAME
      DIMENSION WA(4)

      DO10 K=1,KMAX
      S=0.
      NP=0
      IP=0

      DO20 J=1,IMJM

      R=HA(J,K)

      IF(HM(J,K).NE.0..AND.HG(J).NE.0.)THEN

      NP=NP+1
```



```
      S=S+R

         IF(IP.NE.0)THEN

         IF(R.GT.RMX)RMX=R
         IF(R.LT.RMN)RMN=R

         ELSE

         RMX=R
         RMN=R
         IP=1

         ENDIF

      ENDIF

 20       CONTINUE   !   J

      IF(NP.NE.0)THEN
      WA(1)=S/FLOAT(NP)
      WA(2)=RMN
      WA(3)=RMX
      WA(4)=FLOAT(NP)
      ELSE
      WA(1)=-999.0
      WA(2)=-999.0
      WA(3)=-999.0
      WA(4)=0.
      ENDIF

      IF(ICL.EQ.1)THEN
      IF(K.EQ.1)THEN
      WRITE(NU,*)'EXTR2M: name=',NAME
      WRITE(NU,101)'Level','Mean','Min','Max','Points'
      ENDIF

      WRITE(NU,100)K,(WA(I),I=1,3),INT(WA(4))
      ENDIF
 10       CONTINUE   !   K
      RETURN
 100     FORMAT(I3,3X,3(G10.4,1X),I6)
 101        FORMAT(A5,1X,3A11,1X,A6)
      END
C& PASSAB
      SUBROUTINE PASSAB(A,M,B)
      DIMENSION A(M), B(M)
      DO J=1,M
      B(J)=A(J)
      ENDDO
      RETURN
      END
C& EGRIDS
      SUBROUTINE EGRIDS(DPHID,DLAMD)
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
C
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)

      COMMON/ETAREA/CAREA(IMJM)

      DATA ITMP/ 0,-1, 0, 1, 2,
     *           1, 0,-1,-2,-3,
```

```
     *          -2,-1, 0, 1, 2,
     *           3, 4, 3, 2, 1,
     *           0,-1,-2,-3,-4/
      DATA JTMP/ 0,-1,-2,-1, 0,
     *           1, 2, 1, 0,-1,
     *          -2,-3,-4,-3,-2,
     *          -1, 0, 1, 2, 3,
     *           4, 3, 2, 1, 0/
      DATA ERAD /6370000./
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
CX
CX       DPHID=15./57.        !   in degrees for 40km
CX       DLAMD=5./18.         !   in degrees for 40km
CX
       DPHI=DPHID*RADDEG   !   in radian
       DLAM=DLAMD*RADDEG   !   in radian

       DO J=1,JM
       EAT=(J-JM2)*DPHI    !   in radian
       COSETA(J)=COS(EAT)
       YETA(J)=EAT*DEG     !   in degrees

       DO I=1,IMT
       K=INDEXH(I,J,IM)

       IF(K.NE.0)THEN
       CAREA(K)=2.*ERAD*COSETA(J)*DPHI*DLAM*ERAD
       ENDIF

       ENDDO
       ENDDO

       DO I=1,IMT
       EON=(I-IM)*DLAM     !   in radian
       XETA(I)=EON*DEG     !   in degrees
       ENDDO

       XB=(1-IM)*DLAMD     !   in degrees
       DEX=DLAMD           !   in degrees
       YB=(1-JM2)*DPHID    !   in degrees
       DEY=DPHID           !   in degrees

       KTMP(1)=  0
       KTMP(2)=-IM
       KTMP(3)=-2*IM+1
       KTMP(4)=-IM+1
       KTMP(5)=1
       KTMP(6)=IM
       KTMP(7)=2*IM-1
       KTMP(8)=IM-1
       KTMP(9)=-1
       KTMP(10)=-IM-1
       KTMP(11)=-2*IM
       KTMP(12)=-3*IM+1
       KTMP(13)=-4*IM+2
       KTMP(14)=-3*IM+2
       KTMP(15)=-2*IM+2
       KTMP(16)=-IM+2
       KTMP(17)=2
       KTMP(18)=IM+1
       KTMP(19)=2*IM
       KTMP(20)=3*IM-1
       KTMP(21)=4*IM-2
       KTMP(22)=3*IM-2
       KTMP(23)=2*IM-2
       KTMP(24)=IM-2
       KTMP(25)=-2
```



```
      RETURN
      END
C& VERPRF
      SUBROUTINE VERPRF(NU,NAME,XLAB,YLAB,XP,YP,Q,
     * LMH,KMAX,X0,X1,DX)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (IMAX=2*IM-1)

      DIMENSION Q(IMJM,KMAX), LMH(IMJM)
      CHARACTER*(*) NAME,XLAB,YLAB
C-----------------------------------------------------
       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD
     &,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-------------------------------------------------------
CX                              C O M M O N /LOOPS/
CX     & KHLA  (JAM),KHHA  (JAM),KVLA  (JAM),KVHA  (JAM)
CXC
CX     &,KHL2  (JM),KHH2  (JM),KVL2  (JM),KVH2  (JM)
CXC
CX     &,LMH   (IMJM),LMV   (IMJM)
C-------------------------------------------------------
      COMMON/ETAREA/CAREA(IMJM)
C-------------------------------------------------------
      DIMENSION P(LM),QP(LM)
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       GLAT=YP*RADDEG
       GLON=XP*RADDEG
       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

      K=INDXHC(XE,YE)
      KP=LMH(K)
      DO I=1,KP
      P(I)=FLOAT(I)
      QP(I)=Q(K,I)
      ENDDO

      MARK=3
      LINE=1
      LITH=3
      CALL GRAPHF(NU,
     *  NAME,XLAB,YLAB,
     * ' ',
     * MARK,LINE,LITH,
     *  QP,P,KP,
     *  X0,X1,DX,
     *  1.,FLOAT(KMAX),1.)
      RETURN
```

```
      END
C& CO2F
      SUBROUTINE CO2F(QFLUX,QFLX,EFIREA)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (IMAX=2*IM-1)
      DIMENSION QFLUX(IMJM),EFIREA(IMJM)

      COMMON/ETAREA/CAREA(IMJM)
C
C     EFIREA(J) in km**2
C     CAREA(J)  in m**2
C
      C1=QFLX*1.E6   ! For converting EFIREA(J) to m**2
      C2=C1*1.E6     ! For converting to ppm
      DO J=1,IMJM
      QFLUX(J)=C2*EFIREA(J)/CAREA(J)
      ENDDO
      WRITE(66,*)'   ///////////////////////////'
      WRITE(66,*)'   DISTRIBUTED SOURCES DEBUG'
      WRITE(66,*)'   ///////////////////////////'
      CALL EXTRAC(EFIREA,IMJM,AMEAN,AMIN,AMAX,0.,NULL)
      WRITE(66,*)'EFIREA=',AMEAN,AMIN,AMAX,' km**2'
      CALL EXTRAC(CAREA,IMJM,AMEAN,AMIN,AMAX,0.,NULL)
      WRITE(66,*)'CAREA =',AMEAN,AMIN,AMAX,' m**2'
      C3=1.E-6
      CALL EXTRAC(QFLUX,IMJM,AMEAN,AMIN,AMAX,0.,NULL)
      WRITE(66,*)'QFLUX =',AMEAN*C3,AMIN*C3,AMAX*C3,' kg/m**2/s'
      C4=1.E4*86400.*C3
      WRITE(66,*)'QFLUX =',AMEAN*C4,AMIN*C4,AMAX*C4,' kg/ha/day'
      RETURN
      END
C& CO2FLX
      SUBROUTINE CO2FLX(QFLUX,XP,YP,QFLX,IP,JP,SQFLX)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (IMAX=2*IM-1)

      DIMENSION QFLUX(IMJM)

C-----------------------------------------------------
       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD
     &,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-------------------------------------------------------
      COMMON/ETAREA/CAREA(IMJM)
C-------------------------------------------------------
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       GLAT=YP*RADDEG
       GLON=XP*RADDEG
       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)
```



```
        XE=ELON*DEG
        YE=ELAT*DEG

       K=INDXHC(XE,YE)
       QFLUX(K)=QFLX
       SQFLX=QFLX*CAREA(K)
C
       CALL INDEXS(K,IMAX,IM,IP,JP)
       WRITE(66,*)'SURFACE SOURCE: LON=',XP,' LAT=',YP
       WRITE(66,*)'QFLX        =',86400.*QFLX*1.E+4,
     *                                   ' kg/ha/day'
       WRITE(66,*)'ETA GRID POINT   I=',IP,'  J=',JP
       WRITE(66,*)'INDEX        K=', K
       WRITE(66,*)'ETA COORDINATE  XE=',XE,' YE=',YE
       WRITE(66,*)'CELL AREA      =',CAREA(K),' m**2'
       WRITE(66,*)'               =',CAREA(K)/1.E+6,' km**2'
       WRITE(66,*)'CO2 SOURCE   SQFLX=',SQFLX,' kg/sec/cell'

       RETURN
       END
C& FONCO2
       SUBROUTINE FONCO2(I,J,LAY,N1,Q,HTM,LMH,QCON,
     * RES,PD,DETA,NCELL,DELQM,ALEY)
       PARAMETER (IM=111, JM=221, LM=38,LSM=19)
       PARAMETER (IMJM=IM*JM-JM/2)
       PARAMETER (IMT=IM*2-1)

       PARAMETER (KKMAX=LM)
       PARAMETER (KMIN =LM-KKMAX)

       DIMENSION Q(IMJM,KKMAX), HTM(IMJM,KKMAX),
     * LMH(IMJM),
     * RES(IMJM),PD(IMJM),DETA(KKMAX)

       COMMON/ETAREA/CAREA(IMJM)
C
       DATA G /9.81/
C
       KMAX=KKMAX
       K=INDEXH(I,J,IM)
       IF(K.NE.0)THEN
       KP=K
       I2=I
       J2=J
       ELSE
       KP=INDEXH(I+1,J,IM)
       I2=I+1
       J2=J
       ENDIF

       N2=N1/2
       NFON=0
       TDQ=0.
       ATML=0.

       DO J1=-N2,N2
       DO I1=-N2,N2
       KP1=KP+J1*IM+I1
       DS=CAREA(KP1)
C ---------------------------------------
C      The source holds constant
C      level of concentration
C ---------------------------------------
       KK=LMH(KP1)
       QQ=Q(KP1,KK)*HTM(KP1,KK)

       IF(QQ.LE.QCON)THEN
       Q(KP1,KK)=QCON*HTM(KP1,KK)
       DQ=(QCON-QQ)*HTM(KP1,KK)
```

```
      DM=HTM(KP1,KK)*DETA(KK)*RES(KP1)*PD(KP1)*DS
      NFON=NFON+INT(HTM(KP1,KK))
      TDQ=TDQ+DQ*DM
      ATML=ATML+DM
      ENDIF
C ----------------------------------------
      ENDDO    !  I1 loop
      ENDDO    !  J1 loop

      NCELL=NFON
      DELQM=TDQ/G        !   1E6 x kg
      ALEY=ATML/G        !   kg

      RETURN

      END
C& CFILT2
      SUBROUTINE CFILT2(XD,YD,KTOP,KBOT,Q,HTM,KMAX,QC,UNDIS)

      PARAMETER (IM=111, JM=221, LM=38,LSM=19)
      PARAMETER (IMJM=IM*JM-JM/2)
      PARAMETER (IMT=IM*2-1)
      PARAMETER (IMAX=IMT)
      PARAMETER (NPOINT=25)
      DIMENSION Q(IMJM,KMAX), HTM(IMJM,KMAX)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
      LOGICAL UNDIS, UNDIS1
C
C     FOR INNER POINTS
C
C      JH=INDXHC(XD,YD)
      JH=INDEXC(XD,YD)

      DO J=1,9
      K=JH+IOA(J)

      IF(Q(K,KTOP).GT.QC)THEN
      UNDIS=.FALSE.
      RETURN
      ENDIF

      IF(Q(K,KBOT).GT.QC)THEN
      UNDIS=.FALSE.
      RETURN
      ENDIF

      ENDDO

      UNDIS=.TRUE.

      RETURN
      END
C    --------------------------------------------
C    Three-dimensional interpolation for ETA grid
C    --------------------------------------------
C& SETID3
      SUBROUTINE SETID3(XB,DEX,YB,DEY,IM,JM,CON)
      COMMON /ID3COM/XB1,DEX1,YB1,DEY1,IM1,JM1,CON1
      XB1=XB
      DEX1=DEX
      YB1=YB
      DEY1=DEY
      IM1=IM
      JM1=JM
      CON1=CON
```



```
      RETURN
      END
C& D3HPOL
      SUBROUTINE D3HPOL(HG,HMASK,ELEV,IMJM,KMAX,X0,Y0,E0,ICL,HG0,IC)
      DIMENSION HMASK(IMJM,KMAX)
      DIMENSION HG(IMJM,KMAX)
      DIMENSION ACORNER(4)
      DIMENSION ELEV(KMAX)
C     ----------------------------------------------------------
      DIMENSION MSK(4,4,4), AA(4,4,4),
     *          XX(4),YY(4),MMM(4,4),AAY(4,4),AAE(4),
     *          EE(4), MSKE(4)
C23456
      COMMON /ID3COM/XB,DX,YB,DY,IM,JM,CON
      COMMON/COMXYH/XDEP,YDEP,IBAS,JBAS,XBAS,YBAS,XLOC,YLOC,
     *              IROT,JROT,XROT,YROT,XRLOC,YRLOC
      DIMENSION IDX(4,4), JDX(4,4), KDX(4,4)
C
      COMMON/DBGXYZ/IXYZ
C
      DATA XX/-1.414214, 0., 1.414214, 2.828428/
      DATA YY/-1.414214, 0., 1.414214, 2.828428/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
C
      XDEP=X0
      YDEP=Y0
C
C      IMJM=IM*JM-JM/2
C
      IMAX=2*IM-1
      JMAX=JM
C4/05/98      D=C2*DX
      I=INT((X0-XB)/DX)+1              !  BASE INDEX I
      J=INT((Y0-YB)/DY)+1              !  BASE INDEX J
      II=I
      JJ=J
      IPJ=I+J                          !  POINT TYPE: EVEN OR ODD
      XIJ=XB+DX*(I-1)                  !  BASE POINT
      YIJ=YB+DY*(J-1)                  !  BASE POINT
C
      IBAS=I
      JBAS=J
      XBAS=XIJ
      YBAS=YIJ
C
      XLOC=(X0-XIJ)/DX
      YLOC=(Y0-YIJ)/DY
C
      IF(MOD(IPJ,2).EQ.0)THEN
C     -----------------------
C            I+J=EVEN
C     -----------------------
      IF(YLOC.GE.XLOC)THEN
           IROT=I
           JROT=J
           ELSE
           IROT=I+1
           JROT=J-1
           ENDIF
      ELSE
C     -----------------------
C            I+J=ODD
C     -----------------------
           IF(YLOC.LE.(1.-XLOC))THEN
C
           IROT=I
           JROT=J-1
           ELSE
```

```
            IROT=I+1
            JROT=J
            ENDIF
      ENDIF
            XROT=(IROT-1)*DX+XB
            YROT=(JROT-1)*DY+YB
            XRLOC=(X0-XROT)/DX
            YRLOC=(Y0-YROT)/DY
            RX0= C1*(XRLOC+YRLOC)
            RY0=-C1*(XRLOC-YRLOC)
C
      IF(JROT.NE.0)THEN
      K=INDEXH(IROT,JROT,IM)
      ELSE
      K=-IM+(IROT+1)/2+1
      ENDIF

      CALL LOCATE(ELEV,KMAX,E0,KP)
      E01=E0
      IF(KP.GT.KMAX)THEN
      KP=KMAX         !    GOTO500
      E01=ELEV(KMAX)
      ENDIF
      IF(KP.LT.    1)THEN
      KP=1            !    GOTO501
      E01=ELEV(1)
      ENDIF

      DO KL=1,4
      KETA=KP+KL-2
C     ------------------------------------
      IF(KETA.GE.1.AND.KETA.LE.KMAX)THEN
      EE(KL)=ELEV(KETA)
      ELSE
      EE(KL)=CON
          IF(KETA.LT.1)THEN
          EE(KL)=ELEV(1)
          ENDIF
          IF(KETA.GT.KMAX)THEN
          EE(KL)=ELEV(KMAX)
          ENDIF
      ENDIF
C     ------------------------------------
      DO J=1,4
      DO I=1,4
      IDXI=IROT+I-J
      IDXJ=JROT+I+J-4
      IDXK=K+IM*(I+J-4)-J+2

      IDX(I,J)=I-J
      JDX(I,J)=I+J-4
      KDX(I,J)=IM*(I+J-4)-J+2

      IF(IDXI.GE.1.AND.IDXI.LE.IMAX
     *               .AND.
     *   IDXJ.GE.1.AND.IDXJ.LE.JMAX
     *               .AND.
     *   KETA.GE.1.AND.KETA.LE.KMAX)THEN
      AA(I,J,KL)=HG(IDXK,KETA)
      MSK(I,J,KL)=INT(HMASK(IDXK,KETA))
      ELSE
      AA(I,J,KL)=CON
      MSK(I,J,KL)=0
      ENDIF
      ENDDO
      ENDDO
C     ------------------------------------
      ENDDO                    !  KL Loop
```



```
      ICLOC=ICL

      IF(ICLOC.EQ.2)THEN
           WRITE(66,*)' ########## D3HPOL ###########'
           WRITE(66,*)'X0  =',X0,    ' Y0  =',Y0
           WRITE(66,*)'XIJ  =',XIJ,   ' YIJ  =',YIJ
           WRITE(66,*)'XLOC =',XLOC,  ' YLOC =',YLOC
           WRITE(66,*)'XROT =',XROT,  ' YROT =',YROT
           WRITE(66,*)'XRLOC=',XRLOC,' YRLOC=',YRLOC
           WRITE(66,*)'RX0 =',RX0 ,   ' RY0 =',RY0
           WRITE(66,*)'C1  =',C1,     ' C2  =',C2

           WRITE(66,*)' KP =',KP
           WRITE(66,*)' I =',II,   ' J =',JJ
           WRITE(66,*)'IROT=',IROT,' JROT=',JROT
      WRITE(66,*)'          MASK'
      DO J1=4,1,-1
      WRITE(66,103)J1,(MSK(I1,J1,KP),I1=1,4)
      ENDDO
      WRITE(66,*)'   AA(4,4)'
      DO J1=4,1,-1
      WRITE(66,102)J1,(AA(I1,J1,KP),I1=1,4)
      ENDDO
      ENDIF
C
C ------------------------
C
C     INTERPOLATION
C
C ------------------------
      DO KL=1,4    !   KL Loop

      DO J=1,4     !   J Loop

      CALL D1In4P(AA(1,J,KL),XX,MSK(1,J,KL),RX0,AAY(J,KL),MMM(J,KL))

      ENDDO        !   J Loop

      CALL D1In4P(AAY(1,KL),YY,MMM(1,KL),RY0,AAE(KL),MSKE(KL))

      ENDDO        !   KL Loop

      CALL D1In4P(AAE,EE,MSKE,E01,HG0,IC)

      IF(IXYZ.EQ.1)THEN
      WRITE(66,*)' ////////// D3HPOL ///////////'
      DO KL=1,4

      WRITE(66,*)' LEVEL=',KL
      WRITE(66,*)'          MASK'

      DO J1=4,1,-1
      WRITE(66,103)J1,(MSK(I1,J1,KL),I1=1,4)

      ENDDO
      WRITE(66,*)'   AA(4,4)'

      DO J1=4,1,-1
      WRITE(66,102)J1,(AA(I1,J1,KL),I1=1,4)
      ENDDO

      ENDDO

      WRITE(66,*)' AAY(J,K)    MMM(J,K)'
      DO J=4,1,-1
      WRITE(66,104)J,(AAY(J,K),K=1,4),(MMM(J,K),K=1,4)
      ENDDO
104   FORMAT(I4,1X,4(1X,F11.6),2X,4(1X,I1))
      WRITE(66,*)' E01=',E01
```

```
      DO K=1,4
      WRITE(66,*)K,AAE(K),MSKE(K)
      ENDDO


      ENDIF

C        STOP 'id3'
C
C     QUASI MONOTONIC ALGORITHM
C
 66       CONTINUE
      RETURN
CX 500     CONTINUE
CX       STOP 'D3HPOL 500'
CX 501     CONTINUE
CX       STOP 'D3HPOL 501'
 100      FORMAT(I4,1X,4(1X,I3),5X,4(1X,I3),5X,4(1X,I4))
 101         FORMAT(I4,1X,4(1X,F4.0),5X,4(1X,I3))
 102            FORMAT(I4,1X,4(1X,F9.3))
 103               FORMAT(I4,1X,4(1X,I3))
      END
C& D1In4P
      SUBROUTINE D1In4P(A,AX,MSK,X,Y,ICL)
      DIMENSION A(4), AX(4), MSK(4)
      COMMON/COSTAT/NUMI,NUMI0,NUMIS(16)
      PARAMETER (ILEN=100)
C
C     1-----2-----3-----4
C
      IS=1000*(1-MSK(1))+100*(1-MSK(2))+10*(1-MSK(3))+1-MSK(4)
C23456789 123456789 123456789 123456789 123456789 123456789 12345678
C        10        20        30        40        50        60
      IF(
     *IS.EQ.1111.OR.IS.EQ.1110.OR.IS.EQ.111.OR.IS.EQ.110
     *)THEN
C     4) 1111
C     5) 1110
C     6) 0111
C     7) 0110
      Y=-999.0
      ICL=0
      RETURN
      ENDIF
C
      IF(X.LT.AX(2).OR.X.GT.AX(3))GOTO500
C
      IS2=8*(1-MSK(1))+4*(1-MSK(2))+2*(1-MSK(2))+(1-MSK(4))+1
      IF(IS2.GT.16.OR.IS2.LT.0)THEN
      WRITE(66,*)'D1In4P(A,AX,MSK,X,Y,ICL)'
      WRITE(66,*)'IS2=',IS2
      WRITE(66,*)'IS =',IS
      WRITE(66,*)'A =',A
      WRITE(66,*)'AX =',AX
      WRITE(66,*)'MSK=',MSK
      WRITE(66,*)'X =',X
      STOP 'IS2 > 16'
      ENDIF
      NUMI=NUMI+1
      NUMIS(IS2)=NUMIS(IS2)+1
      IF(IS.EQ.0)THEN
C     1) 0
C     Cubic splane interpolation
C     X1   X2   X3   X4    AX(*)
C     F1   F2   F3   F4     A(*)
C          D2   D3
      NUMI0=NUMI0+1
      D2=.5*((A(2)-A(1))/(AX(2)-AX(1))
     * +(A(3)-A(2))/(AX(3)-AX(2)))
```



```
      D3=.5*((A(3)-A(2))/(AX(3)-AX(2))
     * +(A(4)-A(3))/(AX(4)-AX(3)))
      DEL=AX(3)-AX(2)
      XX=X-AX(2)
      DD=A(2)
      CC=D2
      FF1=A(3)-A(2)-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      Y=((AA*XX+BB)*XX+CC)*XX+DD
      ICL=1
      RETURN
      ENDIF

      IF(IS.EQ.1)THEN
C     2) 1
C     x----x----x----1
C    A(1) A(2) A(3) A(4)
C   AX(1) ...
C         D2   D3
C
      D2=.5*((A(2)-A(1))/(AX(2)-AX(1))
     * +(A(3)-A(2))/(AX(3)-AX(2)))
C
      D3=0.                         !
      DEL=AX(3)-AX(2)
      XX=X-AX(2)

      DD=A(2)
      CC=D2
      FF1=A(3)-A(2)-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2

      Y=((AA*XX+BB)*XX+CC)*XX+DD
      ICL=1
      RETURN
      ENDIF

      IF(IS.EQ.1000)THEN
C     3) 1000
C     1----x----x----x
C    A(1) A(2) A(3) A(4)
C   AX(1) ...
C         D2   D3
C
C
      D2=0.                                !
      D3=.5*((A(3)-A(2))/(AX(3)-AX(2))
     * +(A(4)-A(3))/(AX(4)-AX(3)))
      DEL=AX(3)-AX(2)
      XX=X-AX(2)
      DD=A(2)
      CC=D2
      FF1=A(3)-A(2)-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2

      Y=((AA*XX+BB)*XX+CC)*XX+DD
      ICL=1
      RETURN
      ENDIF

      IF(IS.EQ.100.OR.IS.EQ.101.OR.IS.EQ.1100.OR.IS.EQ.1101)THEN
C     8) 0100
C     9) 0101
```

```
C     10) 1100
C     11) 1101
C      x----1----x----x
C     A(1) A(2) A(3) A(4)
C    AX(1) ...
C          D2   D3
C
      DX=X-AX(2)
      DX2=0.5*(AX(3)-AX(2))
          IF(DX.LT.DX2)THEN
          Y=-999.0
          ICL=0
          ELSE
          Y=A(3)
          ICL=1
          ENDIF
      RETURN
      ENDIF

      IF(IS.EQ.10.OR.IS.EQ.11.OR.IS.EQ.1010.OR.IS.EQ.1011)THEN
C    12) 0010
C    13) 0011
C    14) 1010
C    15) 1011
C      x----x----1----x
C     A(1) A(2) A(3) A(4)
C    AX(1) ...
C          D2   D3
C
      DX=X-AX(2)
      DX2=0.5*(AX(3)-AX(2))
          IF(DX.GT.DX2)THEN
          Y=-999.0
          ICL=0
          ELSE
          Y=A(2)
          ICL=1
          ENDIF
          RETURN
      ENDIF
      IF(IS.EQ.1001)THEN
C     16) 1001
      Y=A(2)+(X-AX(2))*(A(3)-A(2))/(AX(3)-AX(2))
      ICL=1
      RETURN
      ENDIF

      WRITE(66,*)'D1In4P: IS=',IS
      STOP 'ERROR IN D1In4P'
 500     CONTINUE
      WRITE(66,*)'ERROR IN D1In4P: argument is out interval'
      WRITE(66,*)'Interval=',AX(2),AX(3),' Argument=',X
      WRITE(66,*)'A =',A
      WRITE(66,*)'AX =',AX
      WRITE(66,*)'MSK=',MSK
      STOP
      END
C& D3VPOL
      SUBROUTINE D3VPOL(HG,HMASK,ELEV,IMJM,KMAX,X0,Y0,E0,ICL,HG0,IC)
C      (X0,Y0,XB,DX,YB,DY,HG,HMASK,IM,JM,CON,ICL,Z0,IC)
C
C      Interpolation for eta grid
C      For V grid
C      28/02/98
C      05/05/98      D3VPOL
C
      DIMENSION HMASK(IMJM,KMAX)
      DIMENSION HG(IMJM,KMAX)
      DIMENSION ACORNER(4)
```



```
      DIMENSION ELEV(KMAX)
C
      DIMENSION MSK(4,4,4), AA(4,4,4),
     *          XX(4),YY(4),MMM(4,4),AAY(4,4),AAE(4),
     *          EE(4), MSKE(4)
      COMMON /ID3COM/XB,DX,YB,DY,IM,JM,CON
      COMMON/COMXYH/XDEP,YDEP,IBAS,JBAS,XBAS,YBAS,XLOC,YLOC,
     *              IROT,JROT,XROT,YROT,XRLOC,YRLOC
      DIMENSION IDX(4,4), JDX(4,4), KDX(4,4)
C
      DATA XX/-1.414214, 0., 1.414214, 2.828428/
      DATA YY/-1.414214, 0., 1.414214, 2.828428/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
C
      IMAX=2*IM-1
      JMAX=JM
      I=INT((X0-XB)/DX)+1              !  BASE INDEX I
      J=INT((Y0-YB)/DY)+1              !  BASE INDEX J
      II=I
      JJ=J
      IPJ=I+J                          !  POINT TYPE: EVEN OR ODD
      XIJ=XB+DX*(I-1)                  !  BASE POINT
      YIJ=YB+DY*(J-1)                  !  BASE POINT
      DXX=X0-XIJ
      DYY=Y0-YIJ
      XLOC=(X0-XIJ)/DX
      YLOC=(Y0-YIJ)/DY
C                                     ! ----------------
      IF(MOD(IPJ,2).EQ.1)THEN         !    For V grid    < MOD(IPJ,2).EQ.0 >
C     -----------------------         ! ----------------
C            I+J=EVEN
C     -----------------------
      IF(YLOC.GE.XLOC)THEN
           IROT=I
           JROT=J
           ELSE
           IROT=I+1
           JROT=J-1
           ENDIF
      ELSE
C     -----------------------
C            I+J=ODD
C     -----------------------
           IF(YLOC.LE.(1.-XLOC))THEN
           IROT=I
           JROT=J-1
           ELSE
           IROT=I+1
           JROT=J
           ENDIF
      ENDIF
           XROT=(IROT-1)*DX+XB
           YROT=(JROT-1)*DY+YB
           XLOC=(X0-XROT)/DX
           YLOC=(Y0-YROT)/DY
           RX0= C1*(XLOC+YLOC)
           RY0=-C1*(XLOC-YLOC)
C
      K=INDEXV(IROT,JROT,IM)
C
      CALL LOCATE(ELEV,KMAX,E0,KP)
      E01=E0
      IF(KP.GT.KMAX)THEN
      KP=KMAX
      E01=ELEV(KMAX)
      ENDIF
C     GOTO500
      IF(KP.LT.   1)THEN
```

```
      KP=1
      E01=ELEV(1)
      ENDIF
C     GOTO501
      DO KL=1,4
      KETA=KP+KL-2
C     ----------------------------------
      IF(KETA.GE.1.AND.KETA.LE.KMAX)THEN
      EE(KL)=ELEV(KETA)
      ELSE
      EE(KL)=CON
          IF(KETA.LT.1)THEN
          EE(KL)=ELEV(1)
          ENDIF
          IF(KETA.GT.KMAX)THEN
          EE(KL)=ELEV(KMAX)
          ENDIF
      ENDIF
C     ----------------------------------
      DO J=1,4
      DO I=1,4
      IDXI=IROT+I-J
      IDXJ=JROT+I+J-4
      IDXK=K+IM*(I+J-4)-J+2
      IF(IDXI.GE.1.AND.IDXI.LE.IMAX
     *              .AND.
     *   IDXJ.GE.1.AND.IDXJ.LE.JMAX
     *              .AND.
     *   KETA.GE.1.AND.KETA.LE.KMAX
     *              .AND.
     *   IDXK.GE.1.AND.IDXK.LE.IMJM)THEN
      AA(I,J,KL)=HG(IDXK,KETA)
      MSK(I,J,KL)=INT(HMASK(IDXK,KETA))
      ELSE
      AA(I,J,KL)=CON
      MSK(I,J,KL)=0
      ENDIF
      ENDDO
      ENDDO
C ------------------------
      ENDDO                    !  KL Loop

      ICLOC=ICL

      IF(ICLOC.EQ.2)THEN
           WRITE(66,*)' ########## XYVPOL ###########'
           WRITE(66,*)'X0 =',X0,   ' Y0 =',Y0
           WRITE(66,*)'XIJ =',XIJ,  ' YIJ =',YIJ
           WRITE(66,*)'XROT=',XROT,' YROT=',YROT
           WRITE(66,*)'XLOC=',XLOC,' YLOC=',YLOC
           WRITE(66,*)' RX0=',RX0 ,' RY0=',RY0

           WRITE(66,*)' I =',II,   ' J =',JJ
           WRITE(66,*)'IROT=',IROT,' JROT=',JROT
           WRITE(66,*)' KP=',KP   ,' KETA=',KETA
      DO KL=1,4
      WRITE(66,*)'      MASK  KL=',KL
         DO J1=4,1,-1
         WRITE(66,103)J1,(MSK(I1,J1,KL),I1=1,4)
         ENDDO
      WRITE(66,*)'   AA(4,4,KL)'
         DO J1=4,1,-1
         WRITE(66,102)J1,(AA(I1,J1,KL),I1=1,4)
         ENDDO
      ENDDO
 100     FORMAT(I4,1X,4(1X,I3),5X,4(1X,I3),5X,4(1X,I4))
 101        FORMAT(I4,1X,4(1X,F4.0),5X,4(1X,I3))
 102           FORMAT(I4,1X,4(1X,F9.3))
 103              FORMAT(I4,1X,4(1X,I3))
```



```
      ENDIF
C ------------------------
C
C     INTERPOLATION
C ------------------------
      DO KL=1,4    !   KL Loop

      DO J=1,4     !   J Loop

      CALL D1In4P(AA(1,J,KL),XX,MSK(1,J,KL),RX0,AAY(J,KL),MMM(J,KL))

      ENDDO        !   J Loop

      CALL D1In4P(AAY(1,KL),YY,MMM(1,KL),RY0,AAE(KL),MSKE(KL))

      ENDDO        !   KL Loop

      CALL D1In4P(AAE,EE,MSKE,E01,HG0,IC)

      RETURN
CX 500     CONTINUE
CX       WRITE(66,*)'ERROR IN D3VPOL 500'
CX       WRITE(66,*)'E0 =',E0
CX       WRITE(66,*)'KP =',KP
CX       WRITE(66,*)'KMAX=',KMAX
CX       DO J1=1,KMAX
CX       WRITE(66,*)J1,ELEV(J1)
CX       ENDDO
CX       STOP 'D3VPOL 500'
CX 501     CONTINUE
CX       WRITE(66,*)'ERROR IN D3VPOL 501'
CX       WRITE(66,*)'E0 =',E0
CX       WRITE(66,*)'KP =',KP
CX       WRITE(66,*)'KMAX=',KMAX
CX       DO J1=1,KMAX
CX       WRITE(66,*)J1,ELEV(J1)
CX       ENDDO
CX       STOP 'D3VPOL 501'
      END
C& LINI1D
      SUBROUTINE LINI1D(XA,A,M,XB,B,IB)
      DIMENSION A(M),XA(M)
C
      X=XB
      CALL LOCATE(XA,M,X,IP)
C
      IF(IP.LE.0)THEN
      IP=1
      GOTO500
      ENDIF
C
 10       CONTINUE
      IF(IP.GE.(M+1))THEN
      IP=M-1
      GOTO501
      ENDIF
 20       CONTINUE
      X1=XA(IP)
      X2=XA(IP+1)
      R1=X-X1
      R2=X-X2
      B=A(IP)+(A(IP+1)-A(IP))*R1/(X2-X1)
      IB=IP
C
      RETURN
 500   PRINT*,'Error in LINI1D, X=',X
      PRINT*,'Interval:',XA(1),':',XA(M)
      GOTO10
 501   PRINT*,'Error in LINI1D, X=',X
```

```
      PRINT*,'Interval:',XA(1),':',XA(M)
      GOTO20
      END
C& D3HLIN
      SUBROUTINE D3HLIN(HG,HMASK,ELEV,IMJM,KMAX,
     * X0,Y0,E0,ICL,HG0,IC,
     * SUPR8,NSUPR8)
      DIMENSION HMASK(IMJM,KMAX)
      DIMENSION HG(IMJM,KMAX)
      DIMENSION ACORNER(4)
      DIMENSION ELEV(KMAX)
      DIMENSION SUPR8(8)
C     ----------------------------------------------------------------
      DIMENSION MSK(2,2,2), AA(2,2,2),
     *          XX(2),YY(2),MMM(2,2),AAY(2,2),AAE(2),
     *          EE(2), MSKE(2)
C23456
      COMMON /ID3COM/XB,DX,YB,DY,IM,JM,CON
      COMMON/COMXYH/XDEP,YDEP,IBAS,JBAS,XBAS,YBAS,XLOC,YLOC,
     *              IROT,JROT,XROT,YROT,XRLOC,YRLOC
      DIMENSION IDX(4,4), JDX(4,4), KDX(4,4)
C
      DATA XX/0., 1.414214/
      DATA YY/0., 1.414214/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA HLX/0.707107/
C
      XDEP=X0
      YDEP=Y0
C
C      IMJM=IM*JM-JM/2
C
      IMAX=2*IM-1
      JMAX=JM

      I=INT((X0-XB)/DX)+1             !  BASE INDEX I
      J=INT((Y0-YB)/DY)+1             !  BASE INDEX J
      II=I
      JJ=J
      IPJ=I+J                         !  POINT TYPE: EVEN OR ODD
      XIJ=XB+DX*(I-1)                 !  BASE POINT
      YIJ=YB+DY*(J-1)                 !  BASE POINT
C
      IBAS=I
      JBAS=J
      XBAS=XIJ
      YBAS=YIJ
C
      XLOC=(X0-XIJ)/DX
      YLOC=(Y0-YIJ)/DY
C
      IF(MOD(IPJ,2).EQ.0)THEN
C     -----------------------
C            I+J=EVEN
C     -----------------------
      IF(YLOC.GE.XLOC)THEN
           IROT=I
           JROT=J
           ELSE
           IROT=I+1
           JROT=J-1
           ENDIF
      ELSE
C     -----------------------
C            I+J=ODD
C     -----------------------
           IF(YLOC.LE.(1.-XLOC))THEN
C
```



```
           IROT=I
           JROT=J-1
           ELSE
           IROT=I+1
           JROT=J
           ENDIF
      ENDIF
           XROT=(IROT-1)*DX+XB
           YROT=(JROT-1)*DY+YB
           XRLOC=(X0-XROT)/DX
           YRLOC=(Y0-YROT)/DY
           RX0= C1*(XRLOC+YRLOC)
           RY0=-C1*(XRLOC-YRLOC)
C
      IF(JROT.NE.0)THEN
      K=INDEXH(IROT,JROT,IM)
      ELSE
      K=-IM+(IROT+1)/2+1
      ENDIF

      CALL LOCATE(ELEV,KMAX,E0,KP)
      E01=E0
      IF(KP.GT.KMAX)THEN
      KP=KMAX
      E01=ELEV(KMAX)
      ENDIF                                !   GOTO500
      IF(KP.LT.   1)THEN
      KP=1
      E01=ELEV(1)
      ENDIF                                !    GOTO501

      NSUPR8=0
      DO KL=1,2

      KETA=KP+KL-1

      IF(KETA.GE.1.AND.KETA.LE.KMAX)THEN
      EE(KL)=ELEV(KETA)
      ELSE
      EE(KL)=CON
      ENDIF

      DO J=1,2
      DO I=1,2
      IDXI=IROT+I-J
      IDXJ=JROT+I+J-2
      IDXK=K+IM*(I+J-2)-J+1

      IDX(I,J)=I-J
      JDX(I,J)=I+J-2
      KDX(I,J)=IM*(I+J-2)-J+1

      IF(IDXI.GE.1.AND.IDXI.LE.IMAX
     *              .AND.
     *   IDXJ.GE.1.AND.IDXJ.LE.JMAX
     *              .AND.
     *   KETA.GE.1.AND.KETA.LE.KMAX)THEN
      AA(I,J,KL)=HG(IDXK,KETA)
      MSK(I,J,KL)=INT(HMASK(IDXK,KETA))
      ELSE
      AA(I,J,KL)=CON
      MSK(I,J,KL)=0
      ENDIF

      ENDDO
      ENDDO

CX      ENDDO
CX      ENDDO
```

```
      ENDDO      ! KL=1,2:  KP, KP+1
C ------------------------
C
C     INTERPOLATION
C
C ------------------------

C//      WRITE(66,*)'START INTERPOLATION'
C//      WRITE(66,*)'X0=',X0,' Y0=',Y0,' II=',II,' JJ=',JJ


      XR1=RX0/1.414214
      YR1=RY0/1.414214

      DO KL=1,2   !   KL Loop

      DO J=1,2    !   J Loop

      IF(MSK(1,J,KL)*MSK(2,J,KL).NE.0.)THEN
      AAY(J,KL)=AA(1,J,KL)*(1.-XR1)+AA(2,J,KL)*XR1
      MMM(J,KL)=1
      NSUPR8=NSUPR8+1
      SUPR8(NSUPR8)= AA(1,J,KL)
      NSUPR8=NSUPR8+1
      SUPR8(NSUPR8)= AA(2,J,KL)
      ELSE
      AAY(J,KL)=CON
      MMM(J,KL)=0.
          IF(MSK(1,J,KL).NE.0.)THEN
               NSUPR8=NSUPR8+1
               SUPR8(NSUPR8)=AA(1,J,KL)
               IF(XR1.LE. HLX  )THEN
               AAY(J,KL)=AA(1,J,KL)
               MMM(J,KL)=1
               ENDIF
          ENDIF
          IF(MSK(2,J,KL).NE.0.)THEN
               NSUPR8=NSUPR8+1
               SUPR8(NSUPR8)=AA(2,J,KL)
               IF(XR1.GE. HLX  )THEN
               AAY(J,KL)=AA(2,J,KL)
               MMM(J,KL)=1
               ENDIF
          ENDIF
      ENDIF

      ENDDO       !   J Loop

      IF(MMM(1,KL)*MMM(2,KL).NE.0)THEN
      AAE(KL)=AAY(1,KL)*(1.-YR1)+AAY(2,KL)*YR1
      MSKE(KL)=1
      ELSE
      AAE(KL)=CON
      MSKE(KL)=0
           IF(MMM(1,KL).NE.0)THEN
                 IF(YR1.LE. HLX)THEN
                 AAE(KL)=AAY(1,KL)
                 MSKE(KL)=1
                 ENDIF
           ENDIF
           IF(MMM(2,KL).NE.0)THEN
                 IF(YR1.GE. HLX)THEN
                 AAE(KL)=AAY(2,KL)
                 MSKE(KL)=1
                 ENDIF
           ENDIF
        ENDIF
      ENDDO       !   KL Loop
```



```
      IF(MSKE(1)*MSKE(2).NE.0)THEN

      DELE=EE(2)-EE(1)
      EEE0=E01/DELE
      HG0=AAE(1)*(1.-EEE0)+AAE(2)*EEE0
      IC =1

      ELSE

      HG0=CON
      IC =0
           IF(MSKE(1).NE.0)THEN
           HG0=AAE(1)
           IC =1
           ENDIF

      ENDIF

CX      CALL D1In4P(AAE,EE,MSKE,E0,HG0,IC)

C       STOP 'id3'
C
C     QUASI MONOTONIC ALGORITHM
C
 66        CONTINUE
      RETURN
CX 500     CONTINUE
CX      STOP 'D3HLIN 500'
CX 501     CONTINUE
CX      STOP 'D3HLIN 501'
 100     FORMAT(I4,1X,4(1X,I3),5X,4(1X,I3),5X,4(1X,I4))
 101        FORMAT(I4,1X,4(1X,F4.0),5X,4(1X,I3))
 102           FORMAT(I4,1X,4(1X,F9.3))
 103              FORMAT(I4,1X,4(1X,I3))
      END
C& LOCATE
      SUBROUTINE LOCATE(XX,N,X,J)
      INTEGER J,N
      REAL X,XX(N)
C
C     XX(J) < X < XX(J+1)
C     IF J=0 OR J=N+1 THEN X IS OUT OF RANGE
C
      INTEGER J1,JM,JU
C     ____________________________________________________
C     Note the use of logical equality .eqv., which is
C     true when its two logical operand are either both
C     true or both false. This relation allows the routine
C     to work for both monotnically increasing and
C     monotonocally decreasing orders of XX(1:N)
C     -----------------------------------------------------
C
      J1=0
      JU=N+1
 10       IF(JU-J1.GT.1)THEN
          JM=(JU+J1)/2
          IF((XX(N).GT.XX(1))
     *        .EQV.
     *        (X.GE.XX(JM)))THEN
          J1=JM
          ELSE
          JU=JM
          ENDIF
       GOTO10
       ENDIF
       J=J1
       RETURN
       END
```

```
C& IEXTRA
      SUBROUTINE IEXTRA(IA,M,AMEAN,IAMIN,IAMAX,ICON,NCON)
      DIMENSION IA(M)
      N=0
      S=0.
      DO J=1,M
      IR=IA(J)
       IF(IR.NE.ICON)THEN
       IAMN=IR
       IAMX=IR
       I=J
       GOTO10
       ENDIF
      ENDDO
      AMEAN=FLOAT(ICON)
      IAMIN=ICON
      IAMAX=ICON
      RETURN
C
 10         CONTINUE
      DO J=1,M
      IR=IA(J)
       IF(IR.NE.ICON)THEN
       S=S+FLOAT(IR)
       N=N+1
         IF(IR.GT.IAMX)THEN
         IAMX=IR
         ENDIF
         IF(IR.LT.IAMN)THEN
         IAMN=IR
         ENDIF
       ENDIF
       ENDDO
C
      NCON=M-N
      IF(N.NE.0)THEN
      AMEAN=S/FLOAT(N)
      ELSE
      AMEAN=FLOAT(ICON)
      ENDIF
      IAMIN=IAMN
      IAMAX=IAMX
      RETURN
      END
C& SAMPLX
      SUBROUTINE SAMPLX(X,STAX)
C
C     Subroutine calculates a sample moments for X
C
C     STAX(1) is length of X sample
C     STAX(2) is X sum
C     STAX(3) is X*X sum
C     STAX(4) is X min
C     STAX(5) is X max
C
      DIMENSION STAX(10)

      IF(INT(STAX(1)).EQ.0)THEN
      STAX(2)=0.
      STAX(3)=0.
      ENDIF

      STAX(1)=STAX(1)+1.

      IF(INT(STAX(1)).EQ.1)THEN
      STAX(4)=X
      STAX(5)=X
      ELSE
         IF(X.LT.STAX(4))STAX(4)=X
```



```
         IF(X.GT.STAX(5))STAX(5)=X
      ENDIF

      STAX(2)=STAX(2)+X
      STAX(3)=STAX(3)+X*X

      RETURN
      END
C& WRISAM
      SUBROUTINE WRISAM(NU,NAME,STAX)
      DIMENSION STAX(10)
      CHARACTER*(*) NAME
C
      IF(INT(STAX(1)).NE.0)THEN
      WRITE(NU,*)'SAMPLX:',NAME,';'
      XMEAN=STAX(2)/STAX(1)
      X2=STAX(3)/STAX(1)
      XSQR =SQRT(X2)
      XSTD =SQRT(ABS(X2-XMEAN**2))
      WRITE(NU,*)'SAMPLX LENGTH=',INT(STAX(1))
      WRITE(NU,*)'MEAN        =',XMEAN
      WRITE(NU,*)'SQUARE MEAN =',XSQR
      WRITE(NU,*)'STANDARD DEV.=',XSTD
      WRITE(NU,*)'MIN         =',STAX(4)
      WRITE(NU,*)'MAX         =',STAX(5)
      ELSE
      WRITE(NU,*)'SAMPLX:',NAME,' is empty'
      ENDIF
      RETURN
      END
C& INDEXP
      INTEGER FUNCTION INDEXP(X,Y,I,J)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMAX=2*IM-1)
      PARAMETER (NPOINT=25)
C
C     INDEXP=0  for V points
C     INDEXP=1  for H points
C
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      I=INT((X-XB)/DEX+0.5)+1
      J=INT((Y-YB)/DEY+0.5)+1
      IPJ=I+J
      IF(MOD(IPJ,2).EQ.0)THEN
      INDEXP=1
      ELSE
      INDEXP=0
      ENDIF
      RETURN
      END
C& INDXHC
      INTEGER FUNCTION INDXHC(X,Y)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMAX=2*IM-1)
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C
      A=DEY/DEX
      B=1.
      C=-DEY
      I=INT((X-XB)/DEX)+1
      J=INT((Y-YB)/DEY)+1
      IPJ=I+J
```

```
      IF(MOD(IPJ,2).EQ.0)THEN
C                                                !      d
C      H point                                   !   o-----x
C                                                !   |\    |
      XC1=XB+(I-1)*DEX                           !   | \ + |
      YC1=YB+(J-1)*DEY                           !   |  \  |
      DX=X-XC1                                   !   | - \ |
      DY=Y-YC1                                   !   |    \|
      y1=-A*DX+DEY                               !   x-----o
         IF(DY.GE.y1)THEN
         INDXHC=INDEXH(I+1,J+1,IM)
         ELSE
         INDXHC=INDEXH(I,J,IM)
         ENDIF
      ELSE
C
C      V point
C
      XC1=XB+(I-1)*DEX
      YC1=YB+(J-1)*DEY
      DX=X-XC1
      DY=Y-YC1
         IF(DY.GE.A*DX)THEN
         INDXHC=INDEXH(I,J+1,IM)
         ELSE
         INDXHC=INDEXH(I+1,J,IM)
         ENDIF
      ENDIF
      RETURN
      END
C& INDXVC
      INTEGER FUNCTION INDXVC(X,Y)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMAX=2*IM-1)
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *       XB,DEX,YB,DEY,
     *       COSETA(JM),XETA(IMAX),YETA(JM),
     *       ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C
      A=DEY/DEX
      B=1.
      C=-DEY
      I=INT((X-XB)/DEX)+1
      J=INT((Y-YB)/DEY)+1
C              J=INT((Y-YB)/DEY+0.0005)+1
      IPJ=I+J
      IF(MOD(IPJ,2).EQ.0)THEN
C                                                !      d
C      H point                                   !   o-----x
C                                                !   |    /|
      XC1=XB+(I-1)*DEX                           !   |   / |
      YC1=YB+(J-1)*DEY                           !   |  /  |
      DX=X-XC1                                   !   | /   |
      DY=Y-YC1                                   !   |/    |
C     y1=-A*DX+DEY                               !   x-----o
      y1=A*DX
         IF(DY.GE.y1)THEN
         INDXVC=INDEXV(I,J+1,IM)
         ELSE
         INDXVC=INDEXV(I+1,J,IM)
         ENDIF
      ELSE
C                                    x------o
C                                    |\     |
C      V point                       | \    |
C                                    |  \   |
C                                    |   \  |
C                                    |    \ |
```



```
C                                    |     \|
C                                    o------x
      XC1=XB+(I-1)*DEX
      YC1=YB+(J-1)*DEY
      DX=X-XC1
      DY=Y-YC1
      y1=-A*DX+DEY
         IF(DY.LE.y1)THEN
         INDXVC=INDEXV(I,J,IM)
         ELSE
         INDXVC=INDEXV(I+1,J+1,IM)
         ENDIF
      ENDIF
      RETURN
      END
C& INDEXC
      INTEGER FUNCTION INDEXC(X,Y)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMAX=2*IM-1)
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *       XB,DEX,YB,DEY,
     *       COSETA(JM),XETA(IMAX),YETA(JM),
     *       ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C
      A=DEY/DEX
      B=1.
      C=-DEY
      I=INT((X-XB)/DEX)+1
      J=INT((Y-YB)/DEY)+1
C              J=INT((Y-YB)/DEY+0.0005)+1
      IPJ=I+J
      IF(MOD(IPJ,2).EQ.0)THEN
C                                                !      d
C      H point                                   !   o-----x
C                                                !   |\    |
      XC1=XB+(I-1)*DEX                           !   | \ + |
      YC1=YB+(J-1)*DEY                           !   |  \  |
      DX=X-XC1                                   !   | - \ |
      DY=Y-YC1                                   !   |    \|
      y1=-A*DX+DEY                               !   x-----o

      IF(DX.LT.0..OR.DY.LT.0.)THEN
      WRITE(66,*)'INDEXC:     y=',y1
      WRITE(66,*)'INDEXC:     DX=',DX
      WRITE(66,*)'INDEXC:     DY=',DY
      WRITE(66,*)'I=',I,' J=',J
      WRITE(66,*)'X    =',X,' XB=',XB
      WRITE(66,*)'Y    =',Y,' YB=',YB
      WRITE(66,*)'XC1   =',XC1
      WRITE(66,*)'YC1   =',YC1
      WRITE(66,*)'(X-XB)/DEX=',(X-XB)/DEX
      WRITE(66,*)'(Y-YB)/DEY=',(Y-YB)/DEY

      STOP
      ENDIF

      IF(DY.GE.y1)THEN
      INDEXC=INDEXH(I+1,J+1,IM)
      ELSE
      INDEXC=INDEXH(I,J,IM)
      ENDIF
      ELSE
C     V point
      XC1=XB+(I-1)*DEX
      YC1=YB+(J-1)*DEY
      DX=X-XC1
      DY=Y-YC1
      IF(DY.GE.A*DX)THEN
```

```
      INDEXC=INDEXH(I,J+1,IM)
      ELSE
      INDEXC=INDEXH(I+1,J,IM)
      ENDIF
      ENDIF
      RETURN
      END
C& XYEPOL
      SUBROUTINE XYEPOL(X0,Y0,XB,DX,YB,DY,HG,IM,JM,CON,ICL,Z0,IC)
C
C     Interpolation for eta grid
C     For H grid
C     28/02/98
C
C     WS Version
C
      DIMENSION HG(1)
      DIMENSION ACORNER(4)
C
      COMMON /AUXARR/XX(4,4),YY(4),MSK(4,4),AA(4,4),AAY(4),MMM(4)
     *              ,RX0,RY0
C
      COMMON /HCOUNT/II,JJ,HLOC(4,4),HNUM(1005)

      COMMON/COMXYH/XDEP,YDEP,IBAS,JBAS,XBAS,YBAS,XLOC,YLOC,
     *              IROT,JROT,XROT,YROT,XRLOC,YRLOC
      DIMENSION IDX(4,4), JDX(4,4), KDX(4,4)
C
      DATA C1/0.7071068/
      DATA C2/1.414214 /
C
      XDEP=X0
      YDEP=Y0
C
      IMJM=IM*JM-JM/2
      IMAX=2*IM-1
      JMAX=JM
      D=C2*DX
      I=INT((X0-XB)/DX)+1            !  BASE INDEX I
      J=INT((Y0-YB)/DY)+1            !  BASE INDEX J
      II=I
      JJ=J
      IPJ=I+J                        !  POINT TYPE: EVEN OR ODD
      XIJ=XB+DX*(I-1)                !  BASE POINT
      YIJ=YB+DY*(J-1)                !  BASE POINT
C
      IBAS=I
      JBAS=J
      XBAS=XIJ
      YBAS=YIJ
C
      XLOC=(X0-XIJ)/DX
      YLOC=(Y0-YIJ)/DY
C
      IF(MOD(IPJ,2).EQ.0)THEN
C     -----------------------
C           I+J=EVEN
C     -----------------------
      IF(YLOC.GE.XLOC)THEN
           IROT=I
           JROT=J
           ELSE
           IROT=I+1
           JROT=J-1
           ENDIF
      ELSE
C     -----------------------
C           I+J=ODD
C     -----------------------
```



```
           IF(YLOC.LE.(1.-XLOC))THEN
C
           IROT=I
           JROT=J-1
           ELSE
           IROT=I+1
           JROT=J
           ENDIF
      ENDIF
           XROT=(IROT-1)*DX+XB
           YROT=(JROT-1)*DY+YB
           XRLOC=(X0-XROT)/DX
           YRLOC=(Y0-YROT)/DY
           RX0= C1*(XRLOC+YRLOC)                   !   11/03/98
           RY0=-C1*(XRLOC-YRLOC)                   !   11/03/98

C
      IF(JROT.NE.0)THEN                            !   11/03/98
      K=INDEXH(IROT,JROT,IM)                       !   11/03/98
      ELSE                                         !   11/03/98
      K=-IM+(IROT+1)/2+1                           !   11/03/98
C           INDEXH(IROT+1,JROT+1,IM)  + 1          !   11/03/98
      ENDIF                                        !   11/03/98

      DO J=1,4
      DO I=1,4
      IDXI=IROT+I-J
      IDXJ=JROT+I+J-4
      IDXK=K+IM*(I+J-4)-J+2

      IDX(I,J)=I-J
      JDX(I,J)=I+J-4
      KDX(I,J)=IM*(I+J-4)-J+2

      IF(IDXI.GE.1.AND.IDXI.LE.IMAX
     *             .AND.
     *   IDXJ.GE.1.AND.IDXJ.LE.JMAX)THEN
      AA(I,J)=HG(IDXK)
      MSK(I,J)=1                            !   INT(HMASK(IDXK))
      HLOC(I,J)=HNUM(IDXK)
      ELSE
      AA(I,J)=CON
      MSK(I,J)=0                            ! 1
      HLOC(I,J)=0
      ENDIF
      ENDDO
      ENDDO
      KEY=0

      IF(ICL.EQ.2)THEN
C -----------------------
           WRITE(66,*)' ########## XYHPOL ###########'
           WRITE(66,*)'X0 =',X0,  ' Y0 =',Y0
           WRITE(66,*)'XIJ =',XIJ, ' YIJ =',YIJ
           WRITE(66,*)'XLOC =',XLOC,' YLOC =',YLOC
           WRITE(66,*)'XROT =',XROT,' YROT =',YROT
           WRITE(66,*)'XRLOC=',XRLOC,' YRLOC=',YRLOC
           WRITE(66,*)' RX0 =',RX0 ,' RY0 =',RY0

           WRITE(66,*)' I =',II,   ' J =',JJ
           WRITE(66,*)'IROT=',IROT,' JROT=',JROT
      WRITE(66,*)'    ADDRESS       MASK'
      DO J1=4,1,-1
      WRITE(66,101)J1,(HLOC(I1,J1),I1=1,4),(MSK(I1,J1),I1=1,4)
      ENDDO
      WRITE(66,*)'   AA(4,4)'
      DO J1=4,1,-1
      WRITE(66,102)J1,(AA(I1,J1),I1=1,4)
      ENDDO
```

```
      ENDIF
C ------------------------
C
C     INTERPOLATION
C
C ------------------------      SUBROUTINE D1In4P(A,AX,MSK,X,Y,ICL)

      DO J=1,4
      CALL D1In4P(AA(1,J),XX,MSK(1,J),RX0,AAY(J),MMM(J))
      ENDDO        !   J loop
      CALL D1In4P(AAY,YY,MMM,RY0,Z0,IC)

      RETURN
100   FORMAT(I4,1X,4(1X,I3),5X,4(1X,I3),5X,4(1X,I4))
101   FORMAT(I4,1X,4(1X,F4.0),5X,4(1X,I3))
102   FORMAT(I4,1X,4(1X,F9.3))
      END
C& XYHPOL
      SUBROUTINE XYHPOL(X0,Y0,XB,DX,YB,DY,HG,HMASK,IM,JM,CON,ICL,Z0,IC)
C
C     Interpolation for eta grid
C     For H grid
C     28/02/98
C
C     WS Version
C
      DIMENSION HMASK(1)
      DIMENSION HG(1)
      DIMENSION ACORNER(4)
C
      COMMON /AUXARR/XX(4,4),YY(4),MSK(4,4),AA(4,4),AAY(4),MMM(4)
     *              ,RX0,RY0
C
      COMMON /HCOUNT/II,JJ,HLOC(4,4),HNUM(1005)

      COMMON/COMXYH/XDEP,YDEP,IBAS,JBAS,XBAS,YBAS,XLOC,YLOC,
     *              IROT,JROT,XROT,YROT,XRLOC,YRLOC
      DIMENSION IDX(4,4), JDX(4,4), KDX(4,4)
C
      DATA XX/-1.414214, 0., 1.414214, 2.828428,
     *        -1.414214, 0., 1.414214, 2.828428,
     *        -1.414214, 0., 1.414214, 2.828428,
     *        -1.414214, 0., 1.414214, 2.828428/
      DATA YY/-1.414214, 0., 1.414214, 2.828428/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
C
      XDEP=X0
      YDEP=Y0
C
      IMJM=IM*JM-JM/2
      IMAX=2*IM-1
      JMAX=JM
      D=C2*DX
      I=INT((X0-XB)/DX)+1             !  BASE INDEX I
      J=INT((Y0-YB)/DY)+1             !  BASE INDEX J
      II=I
      JJ=J
      IPJ=I+J                         !  POINT TYPE: EVEN OR ODD
      XIJ=XB+DX*(I-1)                 !  BASE POINT
      YIJ=YB+DY*(J-1)                 !  BASE POINT
C
      IBAS=I
      JBAS=J
      XBAS=XIJ
      YBAS=YIJ
C
      XLOC=(X0-XIJ)/DX
      YLOC=(Y0-YIJ)/DY
```



```
C
      IF(MOD(IPJ,2).EQ.0)THEN
C     -----------------------
C            I+J=EVEN
C     -----------------------
      IF(YLOC.GE.XLOC)THEN
           IROT=I
           JROT=J
           ELSE
           IROT=I+1
           JROT=J-1
           ENDIF
      ELSE
C     -----------------------
C            I+J=ODD
C     -----------------------
           IF(YLOC.LE.(1.-XLOC))THEN
C
           IROT=I
           JROT=J-1
           ELSE
           IROT=I+1
           JROT=J
           ENDIF
      ENDIF
           XROT=(IROT-1)*DX+XB
           YROT=(JROT-1)*DY+YB
           XRLOC=(X0-XROT)/DX
           YRLOC=(Y0-YROT)/DY
           RX0= C1*(XRLOC+YRLOC)                !   11/03/98
           RY0=-C1*(XRLOC-YRLOC)                !   11/03/98

C
      IF(JROT.NE.0)THEN                         !   11/03/98
      K=INDEXH(IROT,JROT,IM)                    !   11/03/98
      ELSE                                      !   11/03/98
      K=-IM+(IROT+1)/2+1                        !   11/03/98
C            INDEXH(IROT+1,JROT+1,IM) + 1       !   11/03/98
      ENDIF                                     !   11/03/98

      DO J=1,4
      DO I=1,4
      IDXI=IROT+I-J
      IDXJ=JROT+I+J-4
      IDXK=K+IM*(I+J-4)-J+2

      IDX(I,J)=I-J
      JDX(I,J)=I+J-4
      KDX(I,J)=IM*(I+J-4)-J+2

      IF(IDXI.GE.1.AND.IDXI.LE.IMAX
     *               .AND.
     *   IDXJ.GE.1.AND.IDXJ.LE.JMAX)THEN
      AA(I,J)=HG(IDXK)
      MSK(I,J)=INT(HMASK(IDXK))
      HLOC(I,J)=HNUM(IDXK)
      ELSE
      AA(I,J)=CON
      MSK(I,J)=0                              ! 1
      HLOC(I,J)=0
      ENDIF
      ENDDO
      ENDDO
      KEY=0

      IF(ICL.EQ.2)THEN
C ------------------------
           WRITE(66,*)' ########## XYHPOL ###########'
           WRITE(66,*)'X0 =',X0,  ' Y0 =',Y0
```

```
          WRITE(66,*)'XIJ =',XIJ, ' YIJ =',YIJ
          WRITE(66,*)'XLOC=',XLOC,' YLOC=',YLOC
          WRITE(66,*)'XROT=',XROT,' YROT=',YROT
          WRITE(66,*)'XRLOC=',XRLOC,' YRLOC=',YRLOC
          WRITE(66,*)' RX0=',RX0 ,' RY0=',RY0

          WRITE(66,*)' I =',II,  ' J =',JJ
          WRITE(66,*)'IROT=',IROT,' JROT=',JROT
       WRITE(66,*)'    ADDRESS       MASK'
       DO J1=4,1,-1
       WRITE(66,101)J1,(HLOC(I1,J1),I1=1,4),(MSK(I1,J1),I1=1,4)
       ENDDO
       WRITE(66,*)'  AA(4,4)'
       DO J1=4,1,-1
       WRITE(66,102)J1,(AA(I1,J1),I1=1,4)
       ENDDO
       ENDIF
C ------------------------
C
C      INTERPOLATION
C
C ------------------------      SUBROUTINE D1In4P(A,AX,MSK,X,Y,ICL)

      DO J=1,4
      CALL D1In4P(AA(1,J),XX,MSK(1,J),RX0,AAY(J),MMM(J))
      ENDDO       !   J loop
      CALL D1In4P(AAY,YY,MMM,RY0,Z0,IC)

      RETURN
100   FORMAT(I4,1X,4(1X,I3),5X,4(1X,I3),5X,4(1X,I4))
101   FORMAT(I4,1X,4(1X,F4.0),5X,4(1X,I3))
102   FORMAT(I4,1X,4(1X,F9.3))
      END
C& GCOORD
      SUBROUTINE GCOORD(I,J,GLON,GLAT)
C
C     I,J are eta grid indexes
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
C-----------------------------------------------------------------------
                                   C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------------------------------
      REAL LAM0
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C23456
      PHI0=TPH0D*RADDEG
      LAM0=-TLM0D*RADDEG                 ! Minus for ETRANS !
      EON=(I-IM)*DLMD*RADDEG
      EAT=(J-JM2)*DPHD*RADDEG
      CALL ETRANS (EAT,EON,GLAT,GLON,PHI0,LAM0)
      GLON=-GLON*DEG
      GLAT=GLAT*DEG

CX      WRITE(66,*)'GCOORD: TPH0D=',TPH0D,' TLM0D=',TLM0D
CX      WRITE(66,*)'PHI0=',PHI0,' LAM0=',LAM0
```



```
      RETURN
      END
C& ETAGSI
      SUBROUTINE ETAGSI
C
C     ETA GRID PARAMETERS FOR 40 km
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
C-----------------------------------------------------------------------
                              C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------------------
       RADDEG=3.14159265/180.
       DEG=180./3.14159265

       clat=-18.0            ! 40 km
       clon= 56.5            ! 40 km
       DPHID=15./57.         ! 40 km
       DLAMD= 5./18.         ! 40 km

       TLM0D=-clon           ! in degree
       TPH0D= clat           ! in degree
       DLMD=DLAMD            ! in degree
       DPHD=DPHID            ! in degree
c  wbd  western extent of domain in degree
       wbd=(im-1)*dlamd
c  sbd  southern extent of domain in degree
       sbd=(jm/2)*dphid
       RETURN
       END
C& ToETA
      SUBROUTINE ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)
      R1=COS(GLAT)*SIN(GLON-GLON0)
      R2=COS(GLAT0)*COS(GLAT)*COS(GLON-GLON0)+SIN(GLAT0)*SIN(GLAT)

      IF(R2.EQ.0.)THEN
      ELON=0.
      ELSE
      ELON=ATAN(R1/R2)
      ENDIF

      R3=COS(GLAT0)*SIN(GLAT)-SIN(GLAT0)*COS(GLAT)*COS(GLON-GLON0)
      ELAT=ASIN(R3)
      RETURN
      END
C& FromETA
      SUBROUTINE FromETA(ELON,ELAT,GLON,GLAT,GLON0,GLAT0)
      R1=SIN(ELAT)*COS(GLAT0)+COS(ELAT)*SIN(GLAT0)*COS(ELON)
      GLAT=ASIN(R1)
      R2=SIN(ELON)*COS(ELAT)/COS(GLAT0)
      GLON=GLON0+ASIN(R2)
      RETURN
      END
C& ETRANS
      SUBROUTINE  ETRANS(PHI,LAM,LAT,LON,PHI0,LAM0)
      REAL*4 LAT,LON,LAM,LAM0
C ASSUMES RADIANS IS SENT AND RETURNED
C TRANSFORMS FROM ETA LAT AND LON TO EARTH LAT AND LON
```

```
      PI=3.14159265
      A=SIN(PHI)*COS(PHI0)+COS(PHI)*SIN(PHI0)*COS(LAM)
      LAT=ASIN(A)
      B=COS(PHI)*COS(LAM)/(COS(LAT)*COS(PHI0))
      D=B-TAN(LAT)*TAN(PHI0)
      IF (D.GT.1.0) D=1.0
      C=ACOS(D)
      C=SIGN(C,-1.0*LAM)
      LON=LAM0+C
      RETURN
      END
C ***************************************
C          TEST AVERAGE
C ***************************************
C& VECS25
      SUBROUTINE VECS25(AF,IP,JP,F,H,IMJM,KMAX,IM,CON)
      DIMENSION F(IMJM,KMAX),H(IMJM,KMAX),AF(KMAX)
      COMMON/TAB25/IOAV(4),IOA(25),W(26)
      KB=INDEXH(IP,JP,IM)
      IF(KB.NE.0)THEN
      DO KL=1,KMAX
      S=0.
      A=1.5
      DO J=1,25
      K=KB+IOA(J)
      S=S+F(K,KL)*H(K,KL)
      A=A+H(K,KL)
      ENDDO
      WW=W(INT(A))
      AF(KL)=S*WW
      ENDDO
      ELSE
      DO KL=1,KMAX
      AF(KL)=CON
      ENDDO
      ENDIF
      RETURN
      END
C& TABAVE
      SUBROUTINE TABAVE(X,Y,AF,NF,F,H,IMJM,KMAX,CON)
      DIMENSION F(IMJM,KMAX),H(IMJM,KMAX),AF(KMAX),NF(KMAX)
      COMMON/TAB25/IOAV(4),IOA(25),W(26)
      KB=INDEXC(X,Y)
      DO KL=1,KMAX
        S=0.
        A=1.5
        DO J=1,25
        K=KB+IOA(J)
        S=S+F(K,KL)*H(K,KL)
        A=A+H(K,KL)
        ENDDO
      WW=W(INT(A))

      IF(INT(A).EQ.0)THEN
      AF(KL)=CON
      ELSE
      AF(KL)=S*WW
      ENDIF

      NF(KL)=INT(A)
      ENDDO
      RETURN
      END
C& READST
      SUBROUTINE READST(NU,ST,NST)
      CHARACTER*(*) ST
      CHARACTER*80 S80
      CHARACTER*1 S1
10    READ(NU,100)S80
```



```
100   FORMAT(A80)
101   FORMAT(80A1)
      IF(S80(1:1).EQ.'*')GOTO 10
      NST=0
      DO I=1,80
      READ(S80(I:I),101)S1
      IF(S1.EQ.' ')GOTO20
      IF(S1.EQ.'_')S1=' '
      ST(I:I)=S1
      NST=NST+1
      ENDDO
20    CONTINUE
      RETURN
      END
C& READI
      SUBROUTINE READI(NU,IVAL)
      CHARACTER*80 S80
10    READ(NU,100,END=500)S80
      IF(S80(1:1).EQ.'*')GOTO 10
      READ(S80,*)IVAL
      RETURN
100   FORMAT(A80)
500   WRITE(66,*)'ERROR IN READI: ',S80
      STOP
      END
C& READR
      SUBROUTINE READR(NU,VAL)
      CHARACTER*80 S80
10    READ(NU,100,END=500)S80
      IF(S80(1:1).EQ.'*')GOTO 10
      READ(S80,*)VAL
      RETURN
100   FORMAT(A80)
500   WRITE(66,*)'ERROR IN READR: ',S80
      STOP
      END
C& READL
      SUBROUTINE READL(NU,VAL)
      CHARACTER*80 S80
      LOGICAL VAL
10    READ(NU,100,END=500)S80
      IF(S80(1:1).EQ.'*')GOTO 10
      READ(S80,*)VAL
      RETURN
100   FORMAT(A80)
500   WRITE(66,*)'ERROR IN READR: ',S80
      STOP
      END
C& AVV25P
      SUBROUTINE AVV25P(NHZ,XP,YP,U,V,H,IMJM,CON,UP,VP)
      DIMENSION U(IMJM), V(IMJM), H(IMJM)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
      KPOINT=INDXVC(XP,YP)
      IF(KPOINT.NE.0)THEN
      UH=0.
      VH=0.
      NH=1
      DO J=1,25
      K=KPOINT+IOA(J)
      UH=UH+H(K)*U(K)
      VH=VH+H(K)*V(K)
      NH=NH+H(K)
      ENDDO
      UH=UH*WET(NH)
      VH=VH*WET(NH)
      IF(NH.GE.NHZ)THEN                              !   NHZ=6
      UP=UH
      VP=VH
```

```
      ELSE
      UP=CON
      VP=CON
      ENDIF
      ELSE
      UP=CON
      VP=CON
      ENDIF
      RETURN
      END
C& AV25P
      SUBROUTINE AV25P(XP,YP,Q,H,IMJM,CON,QP)
      DIMENSION Q(IMJM), H(IMJM)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
      KPOINT=INDEXC(XP,YP)
      IF(KPOINT.NE.0)THEN
      QH=0.
      NH=1
      DO J=1,25
      K=KPOINT+IOA(J)
      QH=QH+H(K)*Q(K)
      NH=NH+H(K)
      ENDDO
      QH=QH*WET(NH)
CX      IF(NH.GE.13)THEN
CX      IF(NH.GE.1)THEN
      IF(NH.GE.6)THEN
      QP=QH
      ELSE
      QP=CON
      ENDIF
      ELSE
      QP=CON
      ENDIF
      RETURN
      END
C& AVALL
      SUBROUTINE AVALL(Q,H,CON,AQ)
C
C     ALL INNER POINTS
C     Subroutine avereges over all inner points
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)
      PARAMETER (NPOINT=25)
      DIMENSION Q(IMJM),H(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      SS=0.
      QQ=0.
      DO J=1,IPN
      J1=JINP(J)
      DS=COSETA(J1)
      K1=INP(J)
      QQ=QQ+Q(K1)*H(K1)*DS
      SS=SS+H(K1)*DS
      ENDDO
      IF(SS.NE.0.)THEN
      AQ=QQ/SS
      ELSE
      AQ=CON
      ENDIF
```



```
      RETURN
      END
C& AVSUR
      SUBROUTINE AVSUR(Q,H,CON,QI,QA)
C
C     Average over surface
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (NPOINT=25)
      DIMENSION Q(IMJM),H(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      DATA G /9.81/
      DATA ERAD /6370000./
      SS=0.
      QQ=0.

      DO J=1,IPN
      J1=JINP(J)
      DS=COSETA(J1)
      JP=INP(J)
      QQ=QQ+Q(JP)*H(JP)*DS
      SS=SS+DS*H(JP)
      ENDDO

      IF(SS.NE.0.)THEN
      QA=QQ/SS
      QI=QQ
      ELSE
      QA=CON
      QI=CON
      ENDIF
      RETURN
      END
C& CELLAR
      SUBROUTINE CELLAR(CMEAN,CMIN,CMAX)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAREA/CAREA(IMJM)

      S=0.
      AMN=CAREA(INP(1))
      AMX=CAREA(INP(1))

      DO J=1,IPN
      JP=INP(J)
      DS=CAREA(JP)
      IF(DS.LT.AMN)AMN=DS
      IF(DS.GT.AMX)AMX=DS
      S=S+DS
      ENDDO
      CMEAN=S/FLOAT(IPN)
      CMIN=AMN
      CMAX=AMX
      RETURN
      END
C& COLMAS
      SUBROUTINE COLMAS(Q,H,RES,PD,PT,DETA,KMAX,
     *      COLCON,COLAIR)
```

```fortran
C
C     INNER POINTS MASS
C     COLCON   - column mass of q in kg x 1.E+6
C     COLAIR   - column mass of air in kg x 1.E+6
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX),RES(IMJM),DETA(KMAX),
     *          PD(IMJM),
     *          COLCON(IMJM), COLAIR(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAREA/CAREA(IMJM)
C
      DATA G /9.81/
C
      DO J=1,IPN
      JP=INP(J)
      DS=CAREA(JP)

         AV=0.
         AM=0.

         DO K=1,KMAX
         DM=H(JP,K)*DETA(K)*RES(JP)*PD(JP)
         AM=AM+DM
         AV=AV+Q(JP,K)*DM
         ENDDO

      COLCON(J)=AV*DS/G
      COLAIR(J)=AM*DS/G

      ENDDO

      RETURN
      END
C& INMASS
      SUBROUTINE INMASS(Q,H,RES,PD,PT,DETA,KMAX,
     *      TQM,TAM,RQM,TAREA)
C
C     INNER POINTS MASS
C     TQM   - mass of q in kg
C     TAM   - mass of air in kg
C     RQM   - mean mixing ratio = TQM/TAM
C     TAREA - total area of inner points
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX),RES(IMJM),DETA(KMAX),
     *          PD(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAREA/CAREA(IMJM)
C
      DATA G /9.81/
C
      SS=0.
      QQ=0.
      SM=0.
      DO J=1,IPN
      JP=INP(J)
      DS=CAREA(JP)

         AV=0.
         AM=0.
```



```fortran
         DO K=1,KMAX
         DM=H(JP,K)*DETA(K)*RES(JP)*PD(JP)
         AM=AM+DM
         AV=AV+Q(JP,K)*DM
         ENDDO

      QQ=QQ+AV*DS
      SM=SM+AM*DS
      SS=SS+DS

      ENDDO
C
C     TQM,TAM,RQM,TAREA
C
      TQM=QQ/G
      TAM=SM/G
      RQM=TQM/TAM
      TAREA=SS

      RETURN
      END
C& AVTOT
      SUBROUTINE AVTOT(Q,H,RES,DETA,KMAX,CON,QI,QA)
C
C     TOTAL INNER POINTS
C     Subroutine avereges over total inner points
C     and over pressure
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (NPOINT=25)
      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX),RES(IMJM),DETA(KMAX)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      DATA G /9.81/
      DATA ERAD /6370000./

      SS=0.
      QQ=0.
      SM=0.
      DO J=1,IPN
      J1=JINP(J)
      DS=COSETA(J1)
      JP=INP(J)
      AV=0.
      AM=0.
         DO K=1,KMAX
         DM=H(JP,K)*DETA(K)*RES(JP)
         AM=AM+DM
         AV=AV+Q(JP,K)*DM
         ENDDO
      IF(AM.NE.0.)THEN
      QQ=QQ+AV*DS
      SM=SM+AM*DS
      SS=SS+DS
      ENDIF

      ENDDO
      IF(SS.NE.0.)THEN
      QA=QQ/SM
      QI=QQ/G
      ELSE
      QA=CON
```

```
      QI=CON
      ENDIF
      RETURN
      END
C& WRIZOU
      SUBROUTINE WRIZOU(NU,IREC,AA,HTM,KMAX,BB,CON)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AA(IMJM,KMAX),HTM(IMJM,KMAX),
     *          BB(IMAX,JMAX,6)
      DIMENSION PE(LM), WA(LM), MSK(LM),
     *          PB(6)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C-----------------------------------------------------------------------
                                 C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------------------------------
      DATA PB/ 95000., 85000., 70000.,
     *         50000., 30000., 20000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
       DO I=1,IMAX
       XI=-80.0+(I-1)*DLON
       GLON=XI*RADDEG

       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

       IF(
     *   XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
     *                .AND.
     *   YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN
```



```
          DO K=1,KMAX
C      X,Y interpolation
       CALL AV25P(XE,YE,AA(1,K),HTM(1,K),IMJM,CON,WA(K))
          ENDDO

          BB(I,J,1)=0.5*(WA(17)+WA(18))     !   about 1000 hPa
          BB(I,J,2)=0.5*(WA(12)+WA(13))     !   about  950 hPa
          BB(I,J,3)=0.5*(WA(8)+WA(9))      !   about  850 hPa
          BB(I,J,4)=0.5*(WA(6)+WA(7))      !   about  800 hPa
          BB(I,J,5)=0.5*(WA(3)+WA(2))      !   about  725 hPa
          BB(I,J,6)=0.5*(WA(2)+WA(3))      !   about  660 hPa
       ELSE
          BB(I,J,1)=CON    !   about 1000 hPa
          BB(I,J,2)=CON    !   about  950 hPa
          BB(I,J,3)=CON    !   about  850 hPa
          BB(I,J,4)=CON    !   about  800 hPa
          BB(I,J,5)=CON    !   about  725 hPa
          BB(I,J,6)=CON    !   about  660 hPa
       ENDIF


       ENDDO     ! I loop
       ENDDO     ! J loop
C
C23456     FILTR
C
      DO K=1,6
      DO J=1,JMAX
      DO I=1,IMAX
      IF(ABS(BB(I,J,K)).LT.1.E-20)BB(I,J,K)=0.
      ENDDO
      ENDDO
      ENDDO

      DO K=1,6
      IREC=IREC+1
      WRITE(NU,rec=IREC)((BB(I,J,K),I=1,IMAX),J=1,JMAX)
      ENDDO
       RETURN
       END
C& WRIOUT
      SUBROUTINE WRIOUT(NU,IREC,AA,HTM,KMAX,BB,CON)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AA(IMJM,KMAX),HTM(IMJM,KMAX),
     *          BB(IMAX,JMAX,6)
      DIMENSION PE(LM), WA(LM), MSK(LM),
     *          PB(6)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C-----------------------------------------------------------------------
                                 C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------------------------------
      DATA PB/ 95000., 85000., 70000.,
```

```
     *          50000., 30000., 20000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
       DO I=1,IMAX
       XI=-80.0+(I-1)*DLON
       GLON=XI*RADDEG

       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

       IF(
     *   XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
     *                .AND.
     *   YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN

          DO K=1,KMAX
C      X,Y interpolation
       CALL AV25P(XE,YE,AA(1,K),HTM(1,K),IMJM,CON,WA(K))
          ENDDO

          BB(I,J,1)=WA(17)     !   about 1000 hPa
          BB(I,J,2)=WA(12)     !   about  950 hPa
          BB(I,J,3)=WA(8)      !   about  850 hPa
          BB(I,J,4)=WA(6)      !   about  800 hPa
          BB(I,J,5)=WA(3)      !   about  725 hPa
          BB(I,J,6)=WA(1)      !   about  660 hPa
       ELSE
          BB(I,J,1)=CON    !   about 1000 hPa
          BB(I,J,2)=CON    !   about  950 hPa
          BB(I,J,3)=CON    !   about  850 hPa
          BB(I,J,4)=CON    !   about  800 hPa
          BB(I,J,5)=CON    !   about  725 hPa
          BB(I,J,6)=CON    !   about  660 hPa
       ENDIF


       ENDDO     ! I loop
       ENDDO     ! J loop
C
C23456     FILTR
C
      DO K=1,6
      DO J=1,JMAX
      DO I=1,IMAX
      IF(ABS(BB(I,J,K)).LT.1.E-20)BB(I,J,K)=0.
      ENDDO
      ENDDO
      ENDDO
```



```
      DO K=1,6
      IREC=IREC+1
      WRITE(NU,rec=IREC)((BB(I,J,K),I=1,IMAX),J=1,JMAX)
      ENDDO
       RETURN
       END
C& WRIVOU
      SUBROUTINE WRIVOU(NU,IREC,AU,AV,HTM,KMAX,BU,BV,CON)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AU(IMJM,KMAX),AV(IMJM,KMAX),
     *          HTM(IMJM,KMAX),
     *          BU(IMAX,JMAX,6),BV(IMAX,JMAX,6)
      DIMENSION PE(LM), WA1(LM), WA2(LM), MSK(LM),
     *          PB(6)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C-----------------------------------------------------------------------
                              C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------------------
      DATA PB/ 95000., 85000., 70000.,
     *         50000., 30000., 20000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
       DO I=1,IMAX
       XI=-80.0+(I-1)*DLON
       GLON=XI*RADDEG

       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

       IF(
     *   XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
```

```
     *                 .AND.
     *   YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN

          DO K=1,KMAX
C
C      X,Y interpolation for U and V
C      6 is minimum point number
C
       CALL AVV25P(6,XE,YE,AU(1,K),AV(1,K),HTM(1,K),IMJM,
     *       CON,WA1(K),WA2(K))
          ENDDO
CX          DO K=1,KMAX
C      X,Y interpolation for V
CX       CALL AV25P(XE,YE,AV(1,K),HTM(1,K),IMJM,CON,WA2(K))
CX          ENDDO

          DO K=1,KMAX
          UR=WA1(K)
          VR=WA2(K)

          IF(UR.NE.CON.AND.VR.NE.CON)THEN
          CALL UVGLOB(ELON,ELAT,GLON0,GLAT0,UR,VR,UG,VG)
          WA1(K)=UG
          WA2(K)=VG
          ELSE
          WA1(K)=CON
          WA2(K)=CON
          ENDIF

          ENDDO

          BU(I,J,1)=WA1(17)     !   about 1000 hPa
          BU(I,J,2)=WA1(12)     !   about  950 hPa
          BU(I,J,3)=WA1(8)      !   about  850 hPa
          BU(I,J,4)=WA1(6)      !   about  800 hPa
          BU(I,J,5)=WA1(3)      !   about  725 hPa
          BU(I,J,6)=WA1(1)      !   about  660 hPa

          BV(I,J,1)=WA2(17)     !   about 1000 hPa
          BV(I,J,2)=WA2(12)     !   about  950 hPa
          BV(I,J,3)=WA2(8)      !   about  850 hPa
          BV(I,J,4)=WA2(6)      !   about  800 hPa
          BV(I,J,5)=WA2(3)      !   about  725 hPa
          BV(I,J,6)=WA2(1)      !   about  660 hPa
       ELSE
          BU(I,J,1)=CON     !   about 1000 hPa
          BU(I,J,2)=CON     !   about  950 hPa
          BU(I,J,3)=CON     !   about  850 hPa
          BU(I,J,4)=CON     !   about  800 hPa
          BU(I,J,5)=CON     !   about  725 hPa
          BU(I,J,6)=CON     !   about  660 hPa

          BV(I,J,1)=CON     !   about 1000 hPa
          BV(I,J,2)=CON     !   about  950 hPa
          BV(I,J,3)=CON     !   about  850 hPa
          BV(I,J,4)=CON     !   about  800 hPa
          BV(I,J,5)=CON     !   about  725 hPa
          BV(I,J,6)=CON     !   about  660 hPa
       ENDIF


       ENDDO     ! I loop
       ENDDO     ! J loop
C
C23456     FILTR
C
      DO K=1,6
      DO J=1,JMAX
```



```
      DO I=1,IMAX
      IF(ABS(BU(I,J,K)).LT.1.E-10)BU(I,J,K)=0.
      IF(ABS(BV(I,J,K)).LT.1.E-10)BV(I,J,K)=0.
      ENDDO
      ENDDO
      ENDDO

      DO K=1,6
      IREC=IREC+1
      WRITE(NU,rec=IREC)((BU(I,J,K),I=1,IMAX),J=1,JMAX)
      ENDDO

      DO K=1,6
      IREC=IREC+1
      WRITE(NU,rec=IREC)((BV(I,J,K),I=1,IMAX),J=1,JMAX)
      ENDDO

       RETURN
       END
C& WRISUR
      SUBROUTINE WRISUR(NU,IREC,AA,HTM,KMAX,BB,CON)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AA(IMJM),HTM(IMJM,KMAX),
     *          BB(IMAX,JMAX)
      DIMENSION PE(LM), MSK(LM),
     *          PB(6)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C-----------------------------------------------------------------------
                                   C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------------------------------
      DATA PB/ 95000., 85000., 70000.,
     *         50000., 30000., 20000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
```

```
        DO I=1,IMAX
        XI=-80.0+(I-1)*DLON
        GLON=XI*RADDEG

        CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

        XE=ELON*DEG
        YE=ELAT*DEG

        IF(
     *   XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
     *                .AND.
     *   YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN
C       X,Y interpolation
        CALL AV25P(XE,YE,AA,HTM(1,1),IMJM,CON,WA)

           BB(I,J)=WA     !   about 1000 hPa
        ELSE
           BB(I,J)=CON    !   about 1000 hPa
        ENDIF


        ENDDO      ! I loop
        ENDDO      ! J loop
C
C23456     FILTR
C
       DO J=1,JMAX
       DO I=1,IMAX
       IF(ABS(BB(I,J)).LT.1.E-20)BB(I,J)=0.
       ENDDO
       ENDDO

       IREC=IREC+1
       WRITE(NU,rec=IREC)((BB(I,J),I=1,IMAX),J=1,JMAX)
        RETURN
        END
C& UVROTS
       SUBROUTINE UVROTS(GLON,GLAT,GLON0,GLAT0,UG,VG,UR,VR)
C
C       Subroutine transforms the horizontal wind components
C       UG,VG in point (GLON,GLAT) in the rotated system
C       components UR,VR
C
       D1=COS(GLAT0)*SIN(GLAT)-SIN(GLAT0)*COS(GLAT)*COS(GLON-GLON0)
       D=SQRT(1.-D1**2)
       SS=SIN(GLAT0)*SIN(GLON-GLON0)
       CC=COS(GLAT0)*COS(GLAT)
       SSC=SIN(GLAT0)*SIN(GLAT)*COS(GLON-GLON0)
       UR=(UG*(CC+SSC)-VG*SS)/D
       VR=(UG*SS+VG*(CC+SSC))/D
       RETURN
       END
C& UVGLOB
       SUBROUTINE UVGLOB(ELON,ELAT,GLON0,GLAT0,UR,VR,UG,VG)
C
C       Subroutine transforms the horizontal wind components
C       UR,VR in point (ELON,ELAT) in the global system
C       components UG,VG
C
       D1=COS(GLAT0)*SIN(ELAT)+SIN(GLAT0)*COS(ELON)*COS(ELAT)
       D=SQRT(1.-D1**2)
       CC=COS(GLAT0)*COS(ELAT)
       SS=SIN(GLAT0)*SIN(ELON)
       SSC=SIN(GLAT0)*SIN(ELAT)*COS(ELON)
       UG=(UR*(CC-SSC)+VR*SS)/D
       VG=(-UR*SS+VR*(CC-SSC))/D
```



```
        RETURN
        END
C& PMAPC
        SUBROUTINE PMAPC(NU,TEXT,A,M,N,ABAS,DA,CON)
        PARAMETER (MAXM=200)
        DIMENSION A(M,N)
        CHARACTER*(*) TEXT
        CHARACTER*1 STR(MAXM)
        CHARACTER*1 TP(12),TM(12),SB,UT(-12:12)
        DIMENSION NUP(0:12),PART(0:12)
        DIMENSION VAL(3,-12:12)
C
        DATA TP/'0','1','2','3','4','5','6','7','8','9','+','@'/
        DATA TM/'A','B','C','D','E','F','G','H','S','T','-','#'/
C               1    2    3    4    5    6    7    8    9   10   11   12
        DATA UT/'#','-','T','S','H','G','F','E','D','C','B','A',
       *        '.',
       *        '0','1','2','3','4','5','6','7','8','9','+','@'/
C
        WRITE(NU,'(/)')
        WRITE(NU,*)' >>>>>>>>>>>>>>>>>>>>>>>>>>>>> ',TEXT,
       *           ' <<<<<<<<<<<<<<<<<<<<<<<<<<<<<'
C
        IF(M.GT.MAXM)THEN
        WRITE(0,*)'ERROR IN PMAP: the length of the line > ',MAXM
        RETURN
        ENDIF
C
        DO K=1,13
        NUP(K-1)=0
        PART(K-1)=0.
        ENDDO
        NCON=0
        NPNT=0
C
        AMN=A(1,1)
        AMX=A(1,1)
        DO1 J=1,N
        DO1 I=1,M
        R=A(I,J)
        IF(R.GT.AMX)AMX=R
        IF(R.LT.AMN)AMN=R
  1     CONTINUE

            IF(AMN.EQ.AMX)THEN
            WRITE(NU,*)'   The field is constant =',AMX
            RETURN
            ENDIF

        DA1=DA
        WRITE(NU,*)' MIN=',AMN,' MAX=',AMX,' DA=',DA1
        AMN=ABAS

        IT=1
        WRITE(NU,'(5X,200I1)')(MOD(J,10),J=1,M)
C -------------------------------------
        DO2 J=1,N

           DO3 I=1,M
           STR(I)=' '
  3        CONTINUE

           DO4 I=1,M
           R=A(I,J)
           IF(R.EQ.CON)THEN
           SB='~'
           NCON=NCON+1
           ELSE
           NPNT=NPNT+1
```

```
        D=R-AMN
        IF(D.GT.0.)THEN
        IP=INT(D/DA1)+1
          IF(IP.GT.12)IP=12
        ELSE
          IF(D.EQ.0.)THEN
          IP=0
          ELSE
          IP=INT(-D/DA1)+1
            IF(IP.GT.12)IP=12
          IP=-IP
          ENDIF
        ENDIF

        SB=UT(IP)
        NUP(IP)=NUP(IP)+1
        ENDIF

        STR(I)=SB
  4     CONTINUE

      WRITE(NU,100)J,(STR(K),K=1,M)
100   FORMAT(1X,I3,1X,200A1)
  2   CONTINUE
      WRITE(NU,*)' '
      WRITE(NU,'(5X,200I1)')(MOD(J,10),J=1,M)
      WRITE(NU,*)' '
C --------------------------------------
      WRITE(NU,102)UT(-12),-12*DA1+AMN,AMN,UT(0)
      DO J=2,12
      WRITE(NU,103)(J-14)*DA1+AMN,UT(J-13),(J-13)*DA1+AMN,
     *(J-2)*DA1+AMN,UT(J-1),(J-1)*DA1+AMN
      ENDDO
      WRITE(NU,104)12*DA1+AMN,UT(12)
102   FORMAT(10X,9X,2X,A1,2X,F9.3,4X,F9.3,2X,A1)
103   FORMAT(10X,F9.3,2X,A1,2X,F9.3,4X,F9.3,2X,A1,2X,F9.3)
104   FORMAT(10X,9X,5X,9X,4X,F9.3,2X,A1)
C
      WRITE(NU,*)' '
      PARTC=NCON/FLOAT(M*N)
      IF(NPNT.NE.0)THEN
       DO K=1,13
       PART(K-1)=NUP(K-1)/FLOAT(NPNT)
       ENDDO
      ENDIF
C
      WRITE(NU,110)
      DO K=1,12
      WRITE(NU,111)UT(K-1),NUP(K-1),PART(K-1)
C      WRITE(NU,*)' ',UT(K-1),' ',NUP(K),' ',PART(K)
      ENDDO
      WRITE(NU,112)'~',PARTC,CON
110   FORMAT(10X,'DISTRIBUTION OF POINTS')
111   FORMAT(10X,A1,1X,I6,1X,F6.3)
112   FORMAT(10X,A1,8X,      F6.3,1X,' points with ',F7.1)

CX      DO J=1,25
CX      WRITE(NU,101)UT(J-13),(J-13)*DA1+AMN
CX      ENDDO
CX101   FORMAT(1X,A1,2X,F9.3)
      RETURN
      END
C& FILTER
      SUBROUTINE FILTER(A,B,M,AMIN,AMPL,CON)
      DIMENSION A(M),B(M)
      DO J=1,M
      R=A(J)
      IF(R.NE.CON)THEN
```



```
      IF(R.LE.AMIN)THEN
      B(J)=0.
      ELSE
      B(J)=AMPL*R
      ENDIF

      ENDIF
      ENDDO
      RETURN
      END
C& FLUXMS
      SUBROUTINE FLUXMS(Q,HTM,U,V,VTM,DT,DELQ)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)

      DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),
     *          U(IMJM,KMAX),V(IMJM,KMAX),
     *          VTM(IMJM,KMAX)

      COMMON/LATBND/NUSN,LSBND(IMAX),LNBND(IMAX),
     *              NUWE,LWBND(JM),  LEBND(JM),
     *              JSOUTH,JNORTH

      PARAMETER (NPOINT=25)

C-----------------------------------------------------------
       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,UETA(IMJM,LM),VETA(IMJM,LM),
     & TETA(IMJM,LM),QETA(IMJM,LM)
C------------------------------------------------------------


      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)

      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
      DATA G/9.81/
      DATA PI/3.14159265/
      DATA ERAD /6370000./
C
      CON=PI/180.
C
C     Latitude and longitude steps are converted in radians
C
      DX1=2.*CON*DEX
      DY1=2.*CON*DEY
C
C     WEST AND EAST
C


      DQW=0.
      DQE=0.
      DO L=1,KMAX
```

```
      DO J=1,NUWE

      K=LWBND(J)
      UW=U(K,L)*VTM(K,L)
      DP= DETA(L)*PD(K)*RES(K)
      AIR=DP/G

      K1=K+IM
      K2=K-IM+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)+Q(K2,L))*WET(IW)
      DQW=DQW+UW*Q1*DY1*AIR*ERAD


      K=LEBND(J)
      UE=-U(K,L)*VTM(K,L)
      DP= DETA(L)*PD(K)*RES(K)
      AIR=DP/G

      K1=K+IM
      K2=K-IM+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)+Q(K2,L))*WET(IW)
      DQE=DQE+UE*Q1*DY1*AIR*ERAD
      ENDDO
      ENDDO
C
C     SOUTH AND NORTH
C
      DQS=0.
      DQN=0.
      DO L=1,KMAX

      DO J=1,NUWE

      K=LSBND(J)
      VS=V(K,L)*VTM(K,L)
      DP= DETA(L)*PD(K)*RES(K)
      AIR=DP/G

      K1=K
      K2=K+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)+Q(K2,L))*WET(IW)
      DQS=DQS+VS*Q1*DX1*AIR*COSETA(JSOUTH)*ERAD

      K=LNBND(J)
      VN=-V(K,L)*VTM(K,L)
      DP= DETA(L)*PD(K)*RES(K)
      AIR=DP/G

      K1=K
      K2=K+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)+Q(K2,L))*WET(IW)
      DQN=DQN+VN*Q1*DX1*AIR*COSETA(JNORTH)*ERAD

      ENDDO
      ENDDO
      DELQ=DT*(DQW+DQE+DQS+DQN)
      RETURN
      END
C& FLUXV
      SUBROUTINE FLUXV(Q,HTM,WW,DT,DELQ1,DELQ2,UPAR,TOAR)
```



```
C     ----------------------------------
C     Vertical flux througth top boundary
C     ----------------------------------
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)

      DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),
     *          WW(IMJM,KMAX)

      COMMON/ETAREA/CAREA(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)

C-----------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,UETA(IMJM,LM),VETA(IMJM,LM),
     & TETA(IMJM,LM),QETA(IMJM,LM)
C------------------------------------------------------------
C     VFLX=VFLX+Q(J,1)*HTM(J,1)*WW(J,1)*PD(J)*RES(J)*CAREA(J)
C------------------------------------------------------------
      DATA G/9.81/
      DATA PI/3.14159265/
      DATA ERAD /6370000./
C
      CON=PI/180.
      C1=DT/G
      VFLX=0.
      AREA=0.
      VFLXP=0.
      VFLXM=0.
      AREAP=0.
      AREAM=0.
      TAREA=0.
C     ----------------------------
C
C     VM <  0   ^         VP > 0   |
C               |                  v
C     AM =      1.                 0.
C     AP =      0.                 1.
C
C     ----------------------------
      DO J=1,IPN
      KP=INP(J)
      AM=0.5*(1.-SIGN(1.,WW(KP,1)))
      AP=0.5*(1.+SIGN(1.,WW(KP,1)))
      VM=0.5*WW(KP,1)*(1.-SIGN(1.,WW(KP,1)))*HTM(KP,1)
      VP=0.5*WW(KP,1)*(1.+SIGN(1.,WW(KP,1)))*HTM(KP,1)

      VFLX=VFLX+Q(KP,1)*VM*PD(KP)*RES(KP)*CAREA(KP)
      AREA=AREA+HTM(KP,1)*CAREA(KP)*AM
      TAREA=TAREA+HTM(KP,1)*CAREA(KP)
      ENDDO
      DELQ1=C1*VFLX
      UPAR=AREA
      TOAR=TAREA
      IF(AREA.NE.0.)THEN
      DELQ2=DELQ1/AREA
      ELSE
      DELQ2=0.
      ENDIF
      RETURN
      END
C& VLFLUX
      SUBROUTINE VLFLUX(Q,HTM,WW,CW,LEV1,LEV2,DT,VLF,SLF)
C     ----------------------------------
```

```
C      Vertical flux througth top boundary
C      -----------------------------------
       PARAMETER (IM=111,JM=221,LM=38,LSM=19)
       PARAMETER (IMJM = IM*JM - JM/2)
       PARAMETER (LP1=LM+1)
       PARAMETER (IMAX=2*IM-1  , JMAX=JM)
       PARAMETER (KMAX=LM)

       DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),
      *          WW(IMJM,KMAX),
      *          VLF(KMAX),SLF(KMAX)

       COMMON/ETAREA/CAREA(IMJM)
       COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)

C-----------------------------------------------------------
        C O M M O N /VRBLS/
      & PD(IMJM),RES(IMJM),FIS(IMJM)
C
      &,UETA(IMJM,LM),VETA(IMJM,LM),
      & TETA(IMJM,LM),QETA(IMJM,LM)
C------------------------------------------------------------
C      VFLX=VFLX+Q(J,1)*HTM(J,1)*WW(J,1)*PD(J)*RES(J)*CAREA(J)
C------------------------------------------------------------
       DATA G/9.81/
       DATA PI/3.14159265/
       DATA ERAD /6370000./
C      ----------------------------
       CON=PI/180.
       C1=DT/G
C      ----------------------------
       LEVS=LEV2-LEV1+1

       DO KL=1,LEVS
       VLF(KL)=0.
       SLF(KL)=0.
       ENDDO

       DO J=1,IPN
       KP=INP(J)
       DO KL=1,LEVS

       VV=-WW(KP,LEV1+KL-1)*HTM(KP,LEV1+KL-1)*CW

       VLF(KL)=VLF(KL)+
      *    C1*Q(KP,LEV1+KL-1)*VV*PD(KP)*RES(KP)*CAREA(KP)
       SLF(KL)=SLF(KL)+CAREA(KP)*HTM(KP,LEV1+KL-1)

       ENDDO
       ENDDO

       DO KL=1,LEVS
       IF(SLF(KL).NE.0.)THEN
       SLF(KL)=VLF(KL)/SLF(KL)
       ELSE
       SLF(KL)=0.
       ENDIF
       ENDDO

       RETURN
       END
C& COLDIA
       SUBROUTINE COLDIA(NU,ISTEP,COLCON,COLAIR,QCON,ACON)
       PARAMETER (IM=111,JM=221,LM=38,LSM=19)
       PARAMETER (IMJM = IM*JM - JM/2)

       DIMENSION COLCON(IMJM),COLAIR(IMJM)

       COMMON/ETAREA/CAREA(IMJM)
```



```
       COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)

       PARAMETER (INTERV=23)

       DIMENSION X(INTERV),Y(INTERV),QMASS(INTERV),
      *                              NPOIN(INTERV)
       DATA X/0.0,
      *       1.E-12, 1.E-11, 1.E-10, 1.E-9, 1.E-8, 1.E-7,
      *        1.E-6,  1.E-5,  1.E-4, 1.E-3, 1.E-2, 1.E-1,
      *        0.2,    0.3,    0.4,   0.5,   0.6,   0.7,
      *        0.8,    0.9,   1.0,   1.1/
       DO I=1,INTERV
       Y(I)=0.
       QMASS(I)=0.
       NPOIN(I)=0
       ENDDO

       QMIN=COLCON(1)*QCON
       QMAX=COLCON(1)*QCON
       QMEAN=0.
       AMIN=COLAIR(1)*ACON
       AMAX=COLAIR(1)*ACON
       AMEAN=0.

       DO J=1,IPN
       QR=COLCON(J)*QCON
       AR=COLAIR(J)*ACON
       QMEAN=QMEAN+QR
       AMEAN=AMEAN+AR
       IF(QR.GT.QMAX)QMAX=QR
       IF(QR.LT.QMIN)QMIN=QR

       IF(AR.GT.AMAX)AMAX=AR
       IF(AR.LT.AMIN)AMIN=AR
       ENDDO

       QMEAN=QMEAN/FLOAT(IPN)
       AMEAN=AMEAN/FLOAT(IPN)

       WRITE(NU,*)'COLUMN STATISTIC STEP=',ISTEP,' */'
       WRITE(NU,*)'  CO2     &     AIR'
       WRITE(NU,*)'MEAN =',QMEAN,' & ',AMEAN,' */'
       WRITE(NU,*)'MIN =',QMIN ,' & ',AMIN ,' */'
       WRITE(NU,*)'MAX =',QMAX ,' & ',AMAX ,' */'
       WRITE(NU,*)'INNER POINT NUMBER=',IPN,' */'

       IPNEQ0=0
       IPNNE0=0
       DO J=1,IPN
       QR=COLCON(J)*QCON/QMAX
       IF(QR.EQ.0.0)THEN
       Y(1)=Y(1)+1.
       IPNEQ0=IPNEQ0+1
       ELSE
       IPNNE0=IPNNE0+1
       CALL LOCATE(X,INTERV,QR,I)
       Y(I+1)=Y(I+1)+1.
       QMASS(I+1)=QMASS(I+1)+QR
       NPOIN(I+1)=NPOIN(I+1)+1
       ENDIF
       ENDDO

       DO I=1,INTERV
       IF(NPOIN(I).NE.0)THEN
       QMASS(I)=QMASS(I)/FLOAT(NPOIN(I))
       ELSE
       QMASS(I)=0.
       ENDIF
       ENDDO
```

```
      WRITE(NU,*)' COLUMN DISTRIBUTION STEP=',ISTEP,' */'
      WRITE(NU,*)'COLUMN NUMBER WITHOUT CO2=',IPNEQ0
      WRITE(NU,*)'COLUMN NUMBER WITH   CO2=',IPNNE0
      IF(IPNNE0.NE.0)THEN
      DO I=2,INTERV
      Y(I)=Y(I)/FLOAT(IPNNE0)
      WRITE(NU,*)I-1,X(I-1),X(I),Y(I)*100.,QMASS(I),' kg'
      ENDDO
      WRITE(NU,'(A3)')'end'
      ELSE
      WRITE(NU,'(A3)')'end'
      ENDIF

      RETURN
      END
C& POISTA
      SUBROUTINE POISTA(NU,ISTEP,TEXT,Q,H,KMAX)
C
C                INNER POINTS STATISTICA
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      CHARACTER*(*) TEXT
      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAREA/CAREA(IMJM)
C
      DATA G /9.81/
C
      NP=0
      NPP=0
      NMP=0
      DO J=1,IPN
      JP=INP(J)

         DO K=1,KMAX

         IF(H(JP,K).NE.0.)THEN
         NP=NP+1

         IF(Q(JP,K).LT.0.)THEN
         NMP=NMP+1
         ELSE
         NPP=NPP+1
         ENDIF

         ENDIF

         ENDDO

      ENDDO

      WRITE(NU,*)' '
      WRITE(NU,*)'POINT STATISTIC:',TEXT,': STEP=',ISTEP,' */'
      WRITE(NU,*)'NP=',NP,' NPP=',NPP,' NMP=',NMP,' */'

      RETURN
      END
C& IPOSTA
      SUBROUTINE IPOSTA(NU,ISTEP,TEXT,Q,H,KMAX)
C
C                INNER POINTS STATISTICA
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
```



```
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      CHARACTER*(*) TEXT
      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
C
      DATA G /9.81/
C
      NP=0
      SUQ=0.
      ICL=0
      DO J=1,IPN
      JP=INP(J)

         DO K=1,KMAX

         IF(H(JP,K).NE.0.)THEN
         NP=NP+1

         IF(ICL.EQ.0)THEN
         QMN=Q(JP,K)
         QMX=Q(JP,K)
         ICL=1
         ENDIF

         SUQ=SUQ+H(JP,K)*Q(JP,K)
         IF(Q(JP,K).LT.QMN)QMN=Q(JP,K)
         IF(Q(JP,K).GT.QMX)QMX=Q(JP,K)
         ENDIF

         ENDDO        !  K loop

      ENDDO        !  J loop
      IF(NP.NE.0)THEN
      QMEAN=SUQ/FLOAT(NP)
      ELSE
      QMEAN=SUQ
      ENDIF

      WRITE(NU,*)' '
      WRITE(NU,*)'INNER POINT STATISTIC:',TEXT,': STEP=',ISTEP,' */'
      WRITE(NU,*)'NP=',NP,' MEAN=',QMEAN,' MIN=',QMN,' MAX=',QMX,' */'

      RETURN
      END
C& NEGAT
C     NEGAT(QQ(1,1,2),HTM(1,KTOP),KMAX)
      SUBROUTINE NEGAT(QQ,HTM,KMAX)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)

      DIMENSION QQ(IMJM,KMAX),HTM(IMJM,KMAX)

      DO J=1,IMJM
      DO K=1,KMAX
      IF(HTM(J,K).NE.0.)THEN
        IF(QQ(J,K).LT.0.)QQ(J,K)=0.
      ENDIF
      ENDDO
      ENDDO
      RETURN
      END
C& RDFIRE
      SUBROUTINE RDFIRE(NU,IFDAY,ISTEP,EFIREA)
C
C     EFIREA(IMJM) is fire area on Eta grid
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
```

```
      PARAMETER (IMJM= IM*JM - JM/2)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      DIMENSION EFIREA(IMJM,4)
      COMMON/ETAREA/CAREA(IMJM)

      PARAMETER (NFIL = 10)
      COMMON/DIRICT/DIRTAB(NFIL),IDIR(NFIL)
      CHARACTER*80   DIRTAB

      ICL=MOD(ISTEP,30)
      IF(ICL.EQ.1)THEN
        IDAY=ISTEP/30+IFDAY      !   1

      I1=IDIR(3)
        OPEN(NU,FILE=
     * DIRTAB(3)(1:I1)//'fire_areas_eta2.dat',
     * FORM='UNFORMATTED',ACCESS='DIRECT',RECL=IMJM,
     * STATUS='OLD')

CX        OPEN(NU,FILE=
CX      *'/pesqout/sdg04/igor/eta/fires95/fire_areas_eta.dat',
CX      * FORM='UNFORMATTED',ACCESS='DIRECT',RECL=IMJM,
CX      * STATUS='OLD')


      IREC=(IDAY-1)*4
      DO K=1,4
      IREC=IREC+K
      READ(NU,rec=IREC)(EFIREA(I,K),I=1,IMJM)

      WRITE(66,*)'FIRES RECORD=',IREC,' FIRES DAY=',IDAY
Ci      DO I=1,IMJM
Ci      EFIREA(I,K)=EFIREA(I,K)/CAREA(IMJM)
Ci      ENDDO

      ENDDO
      CLOSE(NU)
      ENDIF
      RETURN
      END
CiC& FSCARB
Ci      SUBROUTINE FSCARB(NU,DIRF,EFIREA)
CiC
CiC     EFIREA(IMJM) is fire area on Eta grid
CiC
Ci      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
Ci      PARAMETER (IMJM= IM*JM - JM/2)
Ci      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
Ci
Ci      CHARACTER*(*) DIRF
Ci      DIMENSION EFIREA(IMJM,4)
Ci      DIMENSION XSB(49),YSB(41),ACOS(41),
Ci     *  DFA(49,41)
Ci
Ci      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
CiC
Ci      PARAMETER (NPOINT=25)
CiC
Ci      COMMON/ETAGRD/
Ci     *       XB,DEX,YB,DEY,
Ci     *       COSETA(JM),XETA(IMAX),YETA(JM),
Ci     *       ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
CiC
Ci
Ci      DATA PAI /3.14159265/
Ci      DATA ERAD/6370./
Ci
Ci      PID180=PAI/180.
```



```
Ci      DLON=PID180*1.875
Ci      DLAT=PID180*1.875
Ci
Ci      OPEN(NU,FILE=DIRF,STATUS='OLD')
Ci      READ(NU,*)M,N
Ci      READ(NU,*)XSB
Ci      READ(NU,*)YSB
Ci      READ(NU,*)ACOS
Ci
Ci      DO1000 K=1,4
CiC123456789 12IIIII123456IIII
CiC DAY=         236 TIME=1145
CiC   FORMAT(A12,I5,A6,I4)
Ci      READ(NU,100)IDAY,IHR
Ci100   FORMAT(A12,I5,A6,I4)
Ci      READ(NU,*)DFA
Ci
Ci      S1=0.
Ci      DO J=1,41
Ci      DO I=1,49
Ci      S1=S1+DFA(I,J)
Ci      ENDDO
Ci      ENDDO
Ci
Ci      SM1=0.
Ci      DO J=1,41
Ci      SS=ACOS(J)*DLON*DLAT*ERAD*ERAD
Ci      W=1./SS
Ci      DO I=1,49
Ci      DFA(I,J)=W*DFA(I,J)  ! THE DENSITY OF THE FIRE AREA
Ci      SM1=SM1+SS
Ci      ENDDO
Ci      ENDDO
Ci
Ci      SM1=SM1/FLOAT(41*49)
Ci
Ci      SM2=0.
Ci      DO IP=1,IPN
Ci      JIN=INP(IP)
Ci      I=IINP(IP)
Ci      J=JINP(IP)
Ci      CALL GCOORD(I,J,GLON,GLAT)
Ci      CALL XY3POL(49,41,XSB,YSB,DFA,1,GLON,GLAT,ZPP)
Ci      SS=2.*PID180*PID180*COSETA(J)*DEX*DEY*ERAD*ERAD
Ci      EFIREA(JIN,K)=SS*ZPP
Ci      SM2=SM2+SS
Ci      ENDDO
Ci
Ci      SM2=SM2/FLOAT(IPN)
Ci
Ci      S2=0.
Ci      DO IP=1,IPN
Ci      JIN=INP(IP)
Ci      S2=S2+EFIREA(JIN,K)
Ci      ENDDO
Ci
Ci      IF(K.EQ.1)
Ci     *WRITE(66,*)'SM1=',SM1,' SM2=',SM2,' km**2 SM1/SM2=',SM1/SM2
Ci      WRITE(66,*)'FSCARB: INPUT=',S1  ,' OUTPUT=',S2,' km**2'
Ci1000  CONTINUE    !   K loop from 1 to 4
Ci      RETURN
Ci      END
C& XY3POL
      SUBROUTINE XY3POL(M,N,X,Y,Z,L,XPP,YPP,ZPP)
C *****************************************
C     BICUBIC LOCAL SPLINE INTERPOLATION
C     08.06.96
C     18.06.96 MODIFICATION FOR POINT SET
C *****************************************
```

```
      DIMENSION Z(M,N),X(M),Y(N)
      DIMENSION XPP(L),YPP(L),ZPP(L)
      DIMENSION ZZ(5,5),XY(5,5)
C
      DO1000 KP=1,L
      XP=XPP(KP)
      YP=YPP(KP)
      CALL LOCATE(X,M,XP,IP)
      IF(IP.LE.0.OR.IP.GE.(M+1))GOTO500
      CALL LOCATE(Y,N,YP,JP)
      IF(JP.LE.0.OR.JP.GE.(N+1))GOTO501
C
      I1=IP-1
      J1=JP-1
      IF(I1.LE.0)I1=1
      IF((M-I1).LE.2)I1=M-3
      I1=I1-1
      IF(J1.LE.0)J1=1
      IF((N-J1).LE.2)J1=N-3
      J1=J1-1
C
      DO J=1,4
      DO I=1,4
      ZZ(I,J)=Z(I1+I,J1+J)
      XY(I,J)=X(I1+I)
      ENDDO
      XY(5,J)=XP
      ENDDO
      DO J=1,4
      XY(J,5)=Y(J1+J)
      ENDDO
      XY(5,5)=YP
C  --------------------------------
C     i-1    i     i+1   i+2
C      x-----x-----x-----x   j+2
C      .                 .
C      x     x     x     x   j+1
C      .                 .
C      x     x     x     x   j
C      .                 .
C      x-----x-----x-----x   j-1
C
C  ---------------------------------
C
      DO J=1,5
      F0=ZZ(1,J)
      F1=ZZ(2,J)
      F2=ZZ(3,J)
      F3=ZZ(4,J)
      S0=XY(1,J)
      S1=XY(2,J)
      S2=XY(3,J)
      S3=XY(4,J)
      S =XY(5,J)
      TETA=(S-S1)/(S2-S1)
      IF(TETA.LE.1.)THEN
C
C     S0    S1    S2   S3
C     F0    F1    F2   F3
C           D1    D2
C     x-----x--i--x----x
      D1=.5*((F1-F0)/(S1-S0)+(F2-F1)/(S2-S1))
      D2=.5*((F2-F1)/(S2-S1)+(F3-F2)/(S3-S2))
      ELSE
        IF(TETA.LT.0.)THEN
C     x--i--x-----x-----x
      D1=(F1-F0)/(S1-S0)
      D2=.5*(D1+(F2-F1)/(S2-S1))
      F2=F1
```



```
      S2=S1
      F1=F0
      S1=S0
      TETA=(S-S1)/(S2-S1)
        ELSE
C     x-----x-----x--i--x
C     0     1     2     3
C           0     1     2
      D2=(F3-F2)/(S3-S2)
      D1=.5*((F2-F1)/(S2-S1)+D2)
      F1=F2
      S1=S2
      F2=F3
      S2=S3
      TETA=(S-S1)/(S2-S1)
        ENDIF
      ENDIF
C
      DEL=S2-S1
      XX=S-S1
      D=F1
      C=D1
      FF1=(F2-C*DEL-D)/DEL**3
      FF2=(D2-C)/(3.*DEL*DEL)
      A=-3.*(2.*FF1/3.-FF2)
      B=3.*DEL*(FF1-FF2)
      ZZ(J,5)=((A*XX+B)*XX+C)*XX+D
C
C      XY(J,5)=Y(J1+J)
C
      ENDDO
      ZPH=ZZ(5,5)
C     ---------------------------- 24.06.96
C     QUASI MONOTONE INTERPOLATION
C     ---------------------------- 24.06.96
      Q1=Z(IP,JP)
      Q2=Z(IP+1,JP)
      Q3=Z(IP+1,JP+1)
      Q4=Z(IP,JP+1)
      QMX=AMAX1(Q1,Q2,Q3,Q4)
      QMN=AMIN1(Q1,Q2,Q3,Q4)
      IF(ZPH.GT.QMX)THEN
      ZP=QMX
      ELSE
          IF(ZPH.LT.QMN)THEN
          ZP=QMN
          ELSE
          ZP=ZPH
          ENDIF
      ENDIF

      ZPP(KP)=ZP

1000  CONTINUE
      RETURN
500   PRINT*,'XY3POL: X out range'
      WRITE(66,*)'XY3POL: X out range'
      WRITE(66,*)'XP=',XP,X(1),X(M)
      STOP 'STOP XY3POL, X'
C      RETURN
501   PRINT*,'XY3POL: Y out range'
      WRITE(66,*)'XY3POL: Y out range'
      WRITE(66,*)'XP=',XP,' YP=',YP
      STOP 'STOP XY3POL, Y'
C      RETURN
      END
C& EXTETA
      SUBROUTINE EXTETA(NU,NAME,A,H,HI,IMJM,KMAX)
      DIMENSION A(IMJM,KMAX),H(IMJM,KMAX),HI(IMJM)
```

```
      CHARACTER*(*) NAME
C
      WRITE(NU,*)'EXTETA: ',NAME,';'
      WRITE(NU,*)'IMJM=',IMJM,' KMAX=',KMAX
      PNUM=0
      S1=0.
      S2=0.
      IEXTR=0
      DO J=1,IMJM
      DO K=1,KMAX
      HH=H(J,K)*HI(J)

      IF(INT(HH).EQ.1)THEN
      PNUM=PNUM+1
      R=A(J,K)
      S1=S1+R
      S2=S2+R*R

        IF(IEXTR.EQ.0)THEN
        AMIN=R
        AMAX=R
        IEXTR=1
        ELSE
           IF(R.LT.AMIN)AMIN=R
           IF(R.GT.AMAX)AMAX=R
        ENDIF

      ENDIF

      ENDDO   !  J
      ENDDO   !  K
        IF(PNUM.NE.0.)THEN
        AMEAN=S1/PNUM
        S2=S2/PNUM
        ASQR=SQRT(S2)
        ASTD=SQRT(S2-AMEAN*AMEAN)
C                 'SAMPLX:'
      WRITE(NU,*)'EXTETA LENGTH=',INT(PNUM)
      WRITE(NU,*)'MEAN       =',AMEAN
      WRITE(NU,*)'SQUARE MEAN =',ASQR
      WRITE(NU,*)'STANDARD DEV.=',ASTD
      WRITE(NU,*)'MIN       =',AMIN
      WRITE(NU,*)'MAX       =',AMAX
      ELSE
      WRITE(NU,*)'EXTETA:',NAME,' is empty'
      ENDIF
      RETURN
      END
C& AMWEIG
      SUBROUTINE AMWEIG(AW,HTM,PD,RES,IMJM,KMAX,DETA,W,SCL)
      DIMENSION AW(IMJM,KMAX),HTM(IMJM,KMAX),
     *          PD(IMJM),RES(IMJM),
     *          DETA(KMAX),W(IMJM)
C
C     Atmospheric mass for weights
C
C     AW=SCL*W*DP/G
C
C     For example: SCL=(Earth radius)**2  m**2
C
C     DP     in hPa
C     W      is area of grid cells in m**2
C     SCL    is scale factor.
C            If SCL has dimention in m**2, then
C            AW has dimention in kg
C
      DATA PAI/3.1415926E0/
      DATA   G/9.81/
C
```



```
      CONV=PAI/180.
      C=SCL/G

      DO K=1,KMAX

      DO J=1,IMJM
      DS=W(J)*C
      C1=DS*PD(J)*RES(J)
      DM=C1*DETA(K)*HTM(J,K)
      AW(J,K)=DM
      ENDDO   ! J loop

      ENDDO   ! K loop
      RETURN
      END
C& SCLINP
      SUBROUTINE SCLINP(A,B,H,M,SCL,SIP)
C
C     Scaling inner product
C
      DIMENSION A(M),B(M),H(M)
      S=0.
      DO I=1,M
      S=S+A(I)*B(I)*H(I)
      ENDDO
      SIP=SCL*S
      RETURN
      END
C& CONSV1
      SUBROUTINE CONSV1(UH,UL,W,H,M,ALFA,BETA,C,CSTAR,IR)
      DIMENSION UH(M),UL(M),ALFA(M,2),BETA(M),W(M),H(M)
C
      SUL=0.
      AB=0.

      DO I=1,M
      SUL=SUL+UL(I)*W(I)*H(I)
      AB=AB+ALFA(I,2)*(UH(I)-UL(I))*W(I)*H(I)
      ENDDO

      CSTAR=C-SUL

      IR=0
      IF(AB.EQ.CSTAR)THEN
         DO I=1,M
         ALFA(I,1)=ALFA(I,2)
         ENDDO
      IR=1
      ENDIF

      SIG=1.
      IF(AB.LT.CSTAR)THEN
      SIG=-1.
      CSTAR=-CSTAR
      ENDIF

      DO I=1,M
      BETA(I)=SIG*(UH(I)-UL(I))*W(I)*H(I)
      ENDDO

      RETURN
      END
C& CONSV2
      SUBROUTINE CONSV2(UH,UL,UM,M,ALFA,BETA,IFLAG,CSTAR,ICON)
      DIMENSION UH(M),UL(M),UM(M),ALFA(M,2),BETA(M),IFLAG(M)
C
C     ALFA(M,1)  are alfa
C     ALFA(M,2)  are max alfa
C
```

```
C
C -----------------------
C          S T E P   1
C -----------------------
      BP=0.
      NP=0
      BN=0.
      NN=0

      DO I=1,M
      BETA1=BETA(I)
      IF(BETA1.LE.0.)THEN
        ALFA(I,1)=ALFA(I,2)      !  alfa  = alfa max
        IFLAG(I)=1               !  iflag = 1
        NN=NN+1                  !  number of negative points
        BN=BN+BETA1
      ELSE
        ALFA(I,1)=0.             !  alfa  = 0.
        IFLAG(I)= 0              !  iflag = 0
        NP=NP+1                  !  number of positive and zero points
        BP=BP+BETA1
      ENDIF
      ENDDO
      SUM=BP+BN
      IT=0
C -----------------------
C         STEP 2
C -----------------------
10    CONTINUE
      IT=IT+1
11    CONTINUE

      BNMAX=0.
      BPMAX=0.
      NZERO=0
      NONE=0
      DO I=1,M
      IF(IFLAG(I).EQ.1)THEN
      BNMAX=BNMAX+ALFA(I,1)*BETA(I)
      NONE=NONE+1
      ELSE
      BPMAX=BPMAX+BETA(I)
      NZERO=NZERO+1
      ENDIF
      ENDDO
C
      SURP=CSTAR-BNMAX
C ------------------------
C     IF SUPR < 0 THEN ...
C ------------------------
C -----------------------
C         STEP 3
C -----------------------
      IF(BPMAX.EQ.0.)THEN
      WRITE(66,*)'ERROR IN CONSER: BPMAX=0'
      WRITE(66,*)'SURP=',SURP,' CSTAR=',CSTAR
      WRITE(66,*)'NONE=',NONE,' NZERO=',NZERO
      ICON=0
      DO I=1,M
      UM(I)=UH(I)
      ENDDO
      RETURN
      ENDIF

      AVALFA=SURP/BPMAX
C -----------------------
C         STEP 4
C -----------------------
      IFL=0
```



```
      SAD=0.
      DO I=1,M
         IF(IFLAG(I).EQ.0)THEN
            IF(AVALFA.LE.ALFA(I,2))THEN
            ALFA(I,1)=AVALFA
            ELSE
            ALFA(I,1)=ALFA(I,2)
            IFL=1
            IFLAG(I)=1
              IF(BETA(I).NE.0.)GOTO10
            SAD=SAD+ALFA(I,1)*BETA(I)
            ENDIF
         ENDIF
      ENDDO
      IF(IFL.NE.0)GOTO10
20    CONTINUE
      DO I=1,M
      UM(I)=ALFA(I,1)*UH(I)+(1.-ALFA(I,1))*UL(I)
      ENDDO
      RETURN
      END
C& CONSV3
      SUBROUTINE CONSV3(UH,UL,UM,M,ALFA)
      DIMENSION UH(M),UL(M),UM(M),ALFA(M,2)
      DO I=1,M
      UM(I)=ALFA(I,1)*UH(I)+(1.-ALFA(I,1))*UL(I)
      ENDDO
      RETURN
      END
C& CONMAS
      SUBROUTINE CONMAS(Q,H,RES,PD,PT,DETA,KMAX,QBKG,QMIN,
     * QM0,DQ,QM)
C
C     INNER POINTS MASS
C     QM0   - mass of q in 10**6 kg before adjustment
C     QM    - mass of q in 10**6 kg after adjustment
C     DQ    - adjustment coefficient
C     QBKG  - background of q
C     QMIN  - min
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)

      DIMENSION Q(IMJM,KMAX),H(IMJM,KMAX),RES(IMJM),DETA(KMAX),
     *          PD(IMJM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAREA/CAREA(IMJM)
C
      DATA G /9.81/
C
      SS=0.
      QQ=0.
      SM=0.

      QB=0.
      SMB=0.

      DO J=1,IPN
      JP=INP(J)
      DS=CAREA(JP)

         AV=0.
         AM=0.

         DO K=1,KMAX
         DM=H(JP,K)*DETA(K)*RES(JP)*PD(JP)
         AM=AM+DM
```

```fortran
         Q1=Q(JP,K)
         AV=AV+Q1*DM

         IF(ABS(Q1-QBKG).GT.QMIN)THEN
         QB =QB +Q1*DM*DS
         SMB=SMB+ DM*DS
         ENDIF

         ENDDO

      QQ=QQ+AV*DS
      SM=SM+AM*DS
      SS=SS+DS

      ENDDO

      DQ=0.0
      QQ=QQ/G
      QB =QB/G
      SMB=SMB/G
      QM=QQ
      DELM=QM0-QQ

      WRITE(66,*)'Inside CONMAS QM0=',QM0,' QQ=',QQ
      WRITE(66,*)'Inside CONMAS DELM=',DELM,' SMB=',SMB
      WRITE(66,*)'Inside CONMAS QB =',QB

      IF(SMB.GT.ABS(DELM).AND.SMB.NE.0.)THEN
      DQ=(QM0-QQ)/SMB

      WRITE(66,*)'Inside CONMAS adjust value DQ=',DQ

      QQ=0.
      DO J=1,IPN
      JP=INP(J)

      C1=CAREA(JP)*RES(JP)*PD(JP)

         DO K=1,KMAX
         DM=H(JP,K)*DETA(K)*C1

         Q1=Q(JP,K)
         IF(ABS(Q1-QBKG).GT.QMIN)THEN
         Q(JP,K)=Q1+DQ
         ENDIF

         QQ=QQ+Q(JP,K)*DM
         ENDDO

      ENDDO
      QM=QQ/G
      ENDIF

C

      RETURN
      END
C *********************************************
C                                             *
C            FROM    I3DHV.f                  *
C                                             *
C *********************************************
C ***********************************************
C      WARNING: DON'T USE FOR LP1 ARRAYS
C      CHANGE: 30/10/98 IT'S POSIBLE USE LP1 ARRAYS
C              SEE NEW PARAMETER FOR CUBARR AND
C              CUBVAR
C ***********************************************
C& LI3D
```



```fortran
      SUBROUTINE LI3D(XP,YP,ZP,PLEV,KMAX,
     * FF,HTM,FFP,HHP)
C     ---------------------------------
C     See /users/igor/trj/eta/simul/i3d.f
C     ---------------------------------
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     *          IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
C
      DIMENSION AX(4),AY(4)
      DIMENSION FF(IMJM,KMAX),
     * HTM(IMJM,KMAX),PLEV(KMAX)
      DIMENSION FTMP(4,4,4),
     * HTMP(4,4,4),P(4)
      DIMENSION FY(4,4), HY(4,4),
     * FZ(4), HZ(4)
      DIMENSION WET(3)
      DATA WET/0., 1., 0.5/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA AX/-1.414214, 0.,
     *          1.414214, 2.828428/
      DATA AY/-1.414214, 0.,
     *          1.414214, 2.828428/
C
      CALL LOCATE(PLEV,KMAX,ZP,KP)
      KTOP=KP
      KBOT=KP+1

      CALL CUBARR
     *(XP,YP,FF,HTM,KMAX,KTOP,KBOT,
     * FTMP,HTMP,PLEV,P,
     * XB,YB,DX,DY)

      XLOC=(XP-XB)/DX
      YLOC=(YP-YB)/DY

      X1=0.
      X2=C2
      X= C1*(XLOC+YLOC)
      Y1=0.
      Y2=C2
      Y=-C1*(XLOC-YLOC)
      Z1=0.
      Z2=1.

      Z=(ZP-P(2))/(P(3)-P(2))

      IF(Z.LT.0.)Z=0.
      IF(Z.GT.1.)Z=1.

      DO K=2,3
      DO J=2,3
      F1=FTMP(2,J,K)*HTMP(2,J,K)
      F2=FTMP(3,J,K)*HTMP(3,J,K)
      DXF=HTMP(2,J,K)*HTMP(3,J,K)
     *         *(F2-F1)/(X2-X1)
      H=HTMP(2,J,K)+HTMP(3,J,K)
      W=WET(INT(H)+1)
      FY(J,K)=W*(F1+F2+(2.*X-X1-X2)
     *                      *DXF)
      HY(J,K)=W*H
      ENDDO      !   J
      F1=FY(2,K)*HY(2,K)
      F2=FY(3,K)*HY(3,K)
```

```
      DYF=HY(2,K)*HY(3,K)*(F2-F1)
     *                    /(Y2-Y1)
      H=HY(2,K)+HY(3,K)
      W=WET(INT(H)+1)
      FZ(K)=W*(F1+F2+(2.*Y-Y1-Y2)*DYF)
      HZ(K)=W*H
      ENDDO      !   K

      F1=FZ(2)*HZ(2)
      F2=FZ(3)*HZ(3)
      DZF=HZ(2)*HZ(3)*(F2-F1)/(Z2-Z1)
      H=HZ(2)+HZ(3)
      W=WET(INT(H)+1)
      FFP=W*(F1+F2+(2.*Z-Z1-Z2)*DZF)
      HHP=W*H

      RETURN

      END
C& LI3DSU
      SUBROUTINE LI3DSU(XP,YP,ZP,PLEV,KMAX,
     * FF,HTM,FSUR,NSUR,FFP,HHP)
C     -----------------------------------
C     See /users/igor/trj/eta/simul/i3d.f
C     This subroutine is equivalent of LI3D
C     with additional capturing of 8 corner
C     points: FSUR(8), NSUR is nomasked point
C             number
C             x-----x
C            /     /|
C           x-----x x
C           |     |/
C           x-----x
C     -----------------------------------
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     *          IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
C
      DIMENSION AX(4),AY(4)
      DIMENSION FF(IMJM,KMAX),
     * HTM(IMJM,KMAX),PLEV(KMAX),
     * FSUR(8)
      DIMENSION FTMP(4,4,4),
     * HTMP(4,4,4),P(4)
      DIMENSION FY(4,4), HY(4,4),
     * FZ(4), HZ(4)
      DIMENSION WET(3)
      DATA WET/0., 1., 0.5/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA AX/-1.414214, 0.,
     *         1.414214, 2.828428/
      DATA AY/-1.414214, 0.,
     *         1.414214, 2.828428/
C
      CALL LOCATE(PLEV,KMAX,ZP,KP)
      KTOP=KP
      KBOT=KP+1

      CALL CUBARR
     *(XP,YP,FF,HTM,KMAX,KTOP,KBOT,
     * FTMP,HTMP,PLEV,P,
     * XB,YB,DX,DY)

      XLOC=(XP-XB)/DX
```



```
      YLOC=(YP-YB)/DY

      X1=0.
      X2=C2
      X= C1*(XLOC+YLOC)
      Y1=0.
      Y2=C2
      Y=-C1*(XLOC-YLOC)
      Z1=0.
      Z2=1.

      Z=(ZP-P(2))/(P(3)-P(2))

      IF(Z.LT.0.)Z=0.
      IF(Z.GT.1.)Z=1.

      DO K=2,3
      DO J=2,3
      F1=FTMP(2,J,K)*HTMP(2,J,K)
      F2=FTMP(3,J,K)*HTMP(3,J,K)
      DXF=HTMP(2,J,K)*HTMP(3,J,K)
     *         *(F2-F1)/(X2-X1)
      H=HTMP(2,J,K)+HTMP(3,J,K)
      W=WET(INT(H)+1)
      FY(J,K)=W*(F1+F2+(2.*X-X1-X2)
     *                      *DXF)
      HY(J,K)=W*H
      ENDDO      !   J
      F1=FY(2,K)*HY(2,K)
      F2=FY(3,K)*HY(3,K)
      DYF=HY(2,K)*HY(3,K)*(F2-F1)
     *                      /(Y2-Y1)
      H=HY(2,K)+HY(3,K)
      W=WET(INT(H)+1)
      FZ(K)=W*(F1+F2+(2.*Y-Y1-Y2)*DYF)
      HZ(K)=W*H
      ENDDO      !   K

      F1=FZ(2)*HZ(2)
      F2=FZ(3)*HZ(3)
      DZF=HZ(2)*HZ(3)*(F2-F1)/(Z2-Z1)
      H=HZ(2)+HZ(3)
      W=WET(INT(H)+1)
      FFP=W*(F1+F2+(2.*Z-Z1-Z2)*DZF)
      HHP=W*H

      I1=0
      DO K=2,3
      DO J=2,3
      DO I=2,3
      H=HTMP(I,J,K)
      IF(H.NE.0.)THEN
      I1=I1+1
      FSUR(I1)=FTMP(I,J,K)
      ENDIF
      ENDDO
      ENDDO
      ENDDO
      NSUR=I1

      RETURN

      END
C& DBGI3D
      SUBROUTINE DBGI3D(NU,NAME,FFP,HHP,
     * AX,AY,P,
     * FTMP,HTMP,FY,HY,FZ,HZ)
      CHARACTER*(*) NAME
      DIMENSION AX(4),AY(4),P(4)
```

```
      DIMENSION FTMP(4,4,4),HTMP(4,4,4)
      DIMENSION FY(4,4),HY(4,4),FZ(4),HZ(4)
      WRITE(NU,*)NAME,' DEBUGING'
      WRITE(NU,*)'RESULT=',FFP,HHP
      DO K=1,4
      DO J=4,1,-1
      WRITE(NU,100)J,
     * (FTMP(I,J,K),I=1,4),
     * (HTMP(I,J,K),I=1,4)
      ENDDO    ! J
      WRITE(NU,*)' '
      ENDDO    ! K   (LEVEL)
      WRITE(NU,*)' '
      WRITE(NU,*)'  FY(J,K)     HY(J,K)'
      DO J=4,1,-1
      WRITE(NU,101)J,
     * (FY(J,K),K=1,4),
     * (HY(J,K),K=1,4)
      ENDDO
      WRITE(NU,*)' '
      WRITE(NU,*)' J, P(4), FZ(4), HZ(4)'
      DO J=1,4
      WRITE(NU,*)J,P(J),FZ(J),HZ(J)
      ENDDO    ! J
      RETURN
100   FORMAT(I3,1X,4(F11.4,1X),3X,4(F4.1,1X))
101   FORMAT(I3,1X,4(1X,F10.6),2X,4(1X,F6.2))
      END
C& CUBI3D
      SUBROUTINE CUBI3D(XP,YP,ZP,PLEV,KMAX,
     * FF,HTM,FFP,HHP)

      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,IMJM=IM*JM-JM/2)
      PARAMETER (NPOINT=25)
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      COMMON/DBGXYZ/IXYZ

      DIMENSION AX(4),AY(4)
      DIMENSION FF(IMJM,KMAX),HTM(IMJM,KMAX),
     *          PLEV(KMAX)
      DIMENSION FTMP(4,4,4),HTMP(4,4,4),P(4)
      DIMENSION FY(4,4),HY(4,4),FZ(4),HZ(4)
      DIMENSION WET(3)

CDB      COMMON/ICOUNT/NUMSTE,NUMPOI,NUMLEV


      DATA WET/0., 1., 0.5/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA AX/-1.414214, 0.,
     *         1.414214, 2.828428/
      DATA AY/-1.414214, 0.,
     *         1.414214, 2.828428/
C
      CALL LOCATE(PLEV,KMAX,ZP,KP)
      KTOP=KP
      KBOT=KP+1

CDB      IF(NUMSTE.EQ.10.AND.NUMPOI.EQ.11346)THEN   !DB
CDB      WRITE(0,*)'CUBI3D: ZP=',ZP,' KTOP=',KTOP,
CDB     *   ' PLEV(1)=',PLEV(1)
CDB      ENDIF

      CALL CUBARR
     *(XP,YP,FF,HTM,KMAX,KTOP,KBOT,
```



```
     * FTMP,HTMP,PLEV,P,
     * XB,YB,DX,DY)

CX       WRITE(66,*)'CUBARR OK, IXYZ=',IXYZ
CX       STOP

      XLOC=(XP-XB)/DX
      YLOC=(YP-YB)/DY

      DX1=AX(2)-AX(1)
      DX2=AX(3)-AX(2)
      DX3=AX(4)-AX(3)
      DY1=AY(2)-AY(1)
      DY2=AY(3)-AY(2)
      DY3=AY(4)-AY(3)
      DZ1=P(2)-P(1)
      DZ2=P(3)-P(2)
      DZ3=P(4)-P(3)

      X= C1*(XLOC+YLOC)
      Y=-C1*(XLOC-YLOC)
      Z=ZP
C
C        F1      F2      F3      F4
C     4  x------x------x------x
C        |      |      |      |
C        |  A1  |  A2  |  A3  |
C     3  x------x------x------x
C        |      W2     W3     |
C        | DX1  | DX2  | DX3  |
C     2  x------x------x------x
C        |      |      |      |
C        |      |      |      |
C     1  x------x------x------x
C        1      2      3      4
C
C


      DO K=1,4
      DO J=1,4
      F1=FTMP(1,J,K)*HTMP(1,J,K)
      F2=FTMP(2,J,K)*HTMP(2,J,K)+(1.-HTMP(2,J,K))*FTMP(3,J,K)
      F3=FTMP(3,J,K)*HTMP(3,J,K)+(1.-HTMP(3,J,K))*FTMP(2,J,K)
      F4=FTMP(4,J,K)*HTMP(4,J,K)
      A1=HTMP(1,J,K)*HTMP(2,J,K)
      A2=HTMP(2,J,K)*HTMP(3,J,K)
      A3=HTMP(3,J,K)*HTMP(4,J,K)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DX1+A2*(F3-F2)/DX2)
      D3=W3*(A2*(F3-F2)/DX2+A3*(F4-F3)/DX3)
      H=HTMP(2,J,K)+HTMP(3,J,K)
      W=WET(INT(H)+1)
      DEL=DX2

      XX=(X-AX(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FY(J,K)=((AA*XX+BB)*XX+CC)*XX+DD
      HY(J,K)=W*H
C     Cubic splane interpolation
C     X1   X2   X3   X4    AX(*)
C     F1   F2   F3   F4     A(*)
C          D2   D3
```

```
      ENDDO     ! ..........................  J
      F1=FY(1,K)*HY(1,K)
      F2=FY(2,K)*HY(2,K)+(1.-HY(2,K))*FY(3,K)
      F3=FY(3,K)*HY(3,K)+(1.-HY(3,K))*FY(2,K)
      F4=FY(4,K)*HY(4,K)
      A1=HY(1,K)*HY(2,K)
      A2=HY(2,K)*HY(3,K)
      A3=HY(3,K)*HY(4,K)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DY1+A2*(F3-F2)/DY2)
      D3=W3*(A2*(F3-F2)/DY2+A3*(F4-F3)/DY3)
      H=HY(2,K)+HY(3,K)
      W=WET(INT(H)+1)
      DEL=DY2

      XX=(Y-AY(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FZ(K)=((AA*XX+BB)*XX+CC)*XX+DD
      HZ(K)=W*H
      ENDDO       ! ............................   K
      F1=FZ(1)*HZ(1)
      F2=FZ(2)*HZ(2)+(1.-HZ(2))*FZ(3)
      F3=FZ(3)*HZ(3)+(1.-HZ(3))*FZ(2)
      F4=FZ(4)*HZ(4)
      A1=HZ(1)*HZ(2)
      A2=HZ(2)*HZ(3)
      A3=HZ(3)*HZ(4)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DZ1+A2*(F3-F2)/DZ2)
      D3=W3*(A2*(F3-F2)/DZ2+A3*(F4-F3)/DZ3)
      H=HZ(2)+HZ(3)
      W=WET(INT(H)+1)
      DEL=DZ2
      XX=(Z-P(2))
CX      /DEL
      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FFP=((AA*XX+BB)*XX+CC)*XX+DD
      HHP=W*H

CX      IF(FFP.GT.10.0)THEN
CXC     .OR.FFP.LT.0.0)THEN

      IF(IXYZ.EQ.1)THEN
      CALL DBGI3D(6,'CUBI3D',FFP,HHP,
     * AX,AY,P,
     * FTMP,HTMP,FY,HY,FZ,HZ)
C      STOP
      ENDIF

CX      WRITE(66,*)'CUBI3D OK'
CX      STOP

      RETURN

      END
C& VEGRID
```



```
      SUBROUTINE VEGRID(PLEV,DP)
C
C     Create vertical mask
C
      PARAMETER (LM=38)
      PARAMETER (KMAX=LM)
      PARAMETER (KMAX5=LM+5)
      DIMENSION PLEV(KMAX)
      COMMON/TMVGRD/IVG(0:KMAX5),
     * HVG(0:KMAX5),
     * EPG(0:KMAX5),DEPG(0:KMAX5)
C
C     IVG   is number level
C     HVG   is mask for level
C     EPG   is eta coordinate on standard levels
C     DEPG  is change of eta coordinate on levels
C
      IVG(0)=1
      HVG(0)=0.
      EPG(0)=PLEV(1)-DP
      DEPG(0)=-DP
      DO K=1,KMAX
      IVG(K)=K
      HVG(K)=1.
      EPG(K)=PLEV(K)
      DEPG(K)=0.
      ENDDO
      DO K=1,5
      IVG(KMAX+K)=KMAX
      HVG(KMAX+K)=0.
      EPG(KMAX+K)=PLEV(KMAX)+K*DP
      DEPG(KMAX+K)=K*DP
      ENDDO

      WRITE(66,*)
     *'VEGRID: LM=',LM,' KMAX=',KMAX
      WRITE(66,*)'    KMAX5=',KMAX5
      WRITE(66,*)
     *' K     IVG(K)    HVG(K)    EPG(K)'
      DO K=0,KMAX5
      WRITE(66,*)K,IVG(K),HVG(K),EPG(K)
      ENDDO

      RETURN
      END
C& CUBARR
      SUBROUTINE CUBARR
     *(X,Y,FF,HTM,KMAX,KTOP,KBOT,FTMP,HTMP,
     * AP,P,XBA,YBA,DXB,DYB)
C
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
CX    ///////////////////////////
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
CX    ///////////////////////////
      DIMENSION FF(IMJM,KMAX),HTM(IMJM,KMAX),
     *          AP(KMAX)
      DIMENSION FTMP(16,4),HTMP(16,4),P(4)
C
      COMMON/TMVGRD/IVG(0:KMAX5),HVG(0:KMAX5),
     * EPG(0:KMAX5),DEPG(0:KMAX5)
      COMMON/ETAGRD/
     * XB,DEX,YB,DEY,
     * COSETA(JM),XETA(IMT),YETA(JM),
     * ITMP(NPOINT),JTMP(NPOINT),KTMP(NPOINT)
CX      DIMENSION MSK(4,4,4), AA(4,4,4),
```

```
CX      * XX(4),YY(4),MMM(4,4),AAY(4,4),AAE(4),
CX      * EE(4), MSKE(4)
C
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
CDB      COMMON/ICOUNT/NUMSTE,NUMPOI,NUMLEV
C
      ITYPE=0

      DX=(X-XB)/DEX
      DY=(Y-YB)/DEY
C
C     Base index
C
      I1=INT(0.5*DX)+1
      J1=INT(0.5*DY)+1

      I=INT(DX)+1
      J=INT(DY)+1
C
C     Local coordinate
C
      X0=XB+(I1-1)*DEX*2.
      Y0=YB+(J1-1)*DEY*2.

      XBQ=X0
      YBQ=Y0
C
C     Base Point Index (address)
C
      K0=(J1-1)*IMT+I1
C
      I1=2*I1-1
      J1=2*J1-1

      IF(I1.EQ.I)THEN
         IF(J1.EQ.J)THEN
C                    I case
            XP=(X-X0)/DEX
            YP=(Y-Y0)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0
            XBA=X0
            YBA=Y0
            ITYPE=10
            ELSE
            KADDR=K0-IM+1
            XBA=X0+DEX
            YBA=Y0-DEY
            ITYPE=11
            ENDIF
         ELSE
C                  III case
            XP=(X-X0)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(XP+YP.LT.1.)THEN
            KADDR=K0
            XBA=X0
            YBA=Y0
            ITYPE=31
            ELSE
            KADDR=K0+IM
            XBA=X0+DEX
            YBA=Y0+DEY
            ITYPE=30
            ENDIF
         ENDIF
        ELSE
         IF(J1.EQ.J)THEN
```



```
C                    II case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0)/DEY
            IF(XP+YP.GE.1.)THEN
            KADDR=K0+1
            XBA=X0+2.*DEX
            YBA=Y0
            ITYPE=20
            ELSE
            KADDR=K0-IM+1
            XBA=X0+DEX
            YBA=Y0-DEY
            ITYPE=21
            ENDIF
         ELSE
C                    IV case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0+IM
            XBA=X0+DEX
            YBA=Y0+DEY
            ITYPE=40
            ELSE
            KADDR=K0+1
            XBA=X0+2.*DEX
            YBA=Y0
            ITYPE=41
            ENDIF

         ENDIF
      ENDIF

CDB      IF(NUMSTE.EQ.10.AND.NUMPOI.EQ.11346)THEN   !DB
CDB      WRITE(0,*)'CUBARR: KTOP=',KTOP
CDB      ENDIF
C
      DO KL=1,4
      KK=KTOP+KL-2
      KKK=IVG(KK)
C      P(KL)=EPG(KK)
      P(KL)=AP(KKK)+DEPG(KK)

      DO J=1,16
      J1=IOAC(J)
      K=KADDR+IOA(J1)
      FTMP(J,KL)=FF(K,KKK)
      HTMP(J,KL)=HTM(K,KKK)*HVG(KK)
      ENDDO     !  J
      ENDDO     !  KL

      DXB=DEX
      DYB=DEY

      RETURN
      END
C& CUBVAR
      SUBROUTINE CUBVAR
     *(X,Y,FF,HTM,KMAX,
     * KTOP,KBOT,FTMP,HTMP,AP,P,
     * XBA,YBA,DXB,DYB)
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     * IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
CX    ///////////////////////////
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
```

```
CX     ///////////////////////////
       DIMENSION FF(IMJM,KMAX),
      * HTM(IMJM,KMAX), AP(KMAX)
       DIMENSION FTMP(16,4), HTMP(16,4), P(4)
C
       COMMON/TMVGRD/IVG(0:KMAX5),HVG(0:KMAX5),
      * EPG(0:KMAX5),DEPG(0:KMAX5)
       COMMON/ETAGRD/
      * XB,DEX,YB,DEY,
      * COSETA(JM),XETA(IMT),YETA(JM),
      * ITMP(NPOINT),JTMP(NPOINT),KTMP(NPOINT)
CX       DIMENSION MSK(4,4,4), AA(4,4,4),
CX      * XX(4),YY(4),MMM(4,4),AAY(4,4),AAE(4),
CX      * EE(4), MSKE(4)
C
       COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
       ITYPE=0

       DX=(X-XB)/DEX
       DY=(Y-YB)/DEY
C
C      Base index
C
       I1=INT(0.5*DX)+1
       J1=INT(0.5*DY)+1

       I=INT(DX)+1
       J=INT(DY)+1
C
C      Local coordinate
C
       X0=XB+(I1-1)*DEX*2.
       Y0=YB+(J1-1)*DEY*2.

       XBQ=X0
       YBQ=Y0
C
C      Base Point Index (address)
C
       K0=(J1-1)*IMT+I1
C
       I1=2*I1-1
       J1=2*J1-1

       IF(I1.EQ.I)THEN
          IF(J1.EQ.J)THEN
C                    I case
             XP=(X-X0)/DEX
             YP=(Y-Y0)/DEY
             IF(YP+YP.GE.1.)THEN
             KADDR=K0
             XBA=X0+DEX
             YBA=Y0
             ITYPE=10
             ELSE
             KADDR=K0-IM
             XBA=X0
             YBA=Y0-DEY
             ITYPE=11
             ENDIF
          ELSE
C                  III case
             XP=(X-X0)/DEX
             YP=(Y-Y0-DEY)/DEY
             IF(YP.GE.XP)THEN
             KADDR=K0+IM-1
             XBA=X0
             YBA=Y0+DEY
```



```
             ITYPE=31
             ELSE
             KADDR=K0
             XBA=X0+DEX
             YBA=Y0
             ITYPE=10
             ENDIF
          ENDIF
        ELSE
          IF(J1.EQ.J)THEN
C                    II case
             XP=(X-X0-DEX)/DEX
             YP=(Y-Y0)/DEY
             IF(YP.GE.XP)THEN
             KADDR=K0
             XBA=X0+DEX
             YBA=Y0
             ITYPE=10
             ELSE
             KADDR=K0-IM+1
             XBA=X0+2.*DEX
             YBA=Y0-DEY
             ITYPE=12
             ENDIF
          ELSE
C                    IV case
             XP=(X-X0-DEX)/DEX
             YP=(Y-Y0-DEY)/DEY
             IF(YP+YP.GE.1.)THEN
             KADDR=K0+IM
             XBA=X0+2.*DEX
             YBA=Y0+DEY
             ITYPE=13
             ELSE
             KADDR=K0
             XBA=X0+DEX
             YBA=Y0
             ITYPE=10
             ENDIF

          ENDIF
       ENDIF
C
       DO KL=1,4
       KK=KTOP+KL-2
       KKK=IVG(KK)
       P(KL)=EPG(KK)

       DO J=1,16
       J1=IOAC(J)
       K=KADDR+IOA(J1)
       FTMP(J,KL)=FF(K,KK)
       HTMP(J,KL)=HTM(K,KK)*HVG(KK)
       ENDDO     !  J
       ENDDO     !  KL

       DXB=DEX
       DYB=DEY

       RETURN
       END
C& VUBI3D
       SUBROUTINE VUBI3D(XP,YP,ZP,PLEV,KMAX,
      * FF,HTM,FFP,HHP)

       PARAMETER (IM=111,JM=221)
       PARAMETER (IMT=2*IM-1,IMJM=IM*JM-JM/2)
       PARAMETER (NPOINT=25)
       COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
```

```
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      COMMON/DBGXYZ/IXYZ

      DIMENSION AX(4),AY(4)
      DIMENSION FF(IMJM,KMAX),HTM(IMJM,KMAX),
     *          PLEV(KMAX)
      DIMENSION FTMP(4,4,4),HTMP(4,4,4),P(4)
      DIMENSION FY(4,4),HY(4,4),FZ(4),HZ(4)
      DIMENSION WET(3)
      DATA WET/0., 1., 0.5/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA AX/-1.414214, 0.,
     *         1.414214, 2.828428/
      DATA AY/-1.414214, 0.,
     *         1.414214, 2.828428/
C
      CALL LOCATE(PLEV,KMAX,ZP,KP)
      KTOP=KP
      KBOT=KP+1

      CALL CUBVAR
     *(XP,YP,FF,HTM,KMAX,KTOP,KBOT,
     * FTMP,HTMP,PLEV,P,
     * XB,YB,DX,DY)


      XLOC=(XP-XB)/DX
      YLOC=(YP-YB)/DY

      DX1=AX(2)-AX(1)
      DX2=AX(3)-AX(2)
      DX3=AX(4)-AX(3)
      DY1=AY(2)-AY(1)
      DY2=AY(3)-AY(2)
      DY3=AY(4)-AY(3)
      DZ1=P(2)-P(1)
      DZ2=P(3)-P(2)
      DZ3=P(4)-P(3)

      X= C1*(XLOC+YLOC)
      Y=-C1*(XLOC-YLOC)
      Z=ZP
C
C        F1      F2      F3      F4
C     4  x------x------x------x
C        |      |      |      |
C        |  A1  |  A2  |  A3  |
C     3  x------x------x------x
C        |      W2     W3     |
C        |  DX1 |  DX2 |  DX3 |
C     2  x------x------x------x
C        |      |      |      |
C        |      |      |      |
C     1  x------x------x------x
C        1      2      3      4
C
C

      DO K=1,4
      DO J=1,4
      F1=FTMP(1,J,K)*HTMP(1,J,K)
      F2=FTMP(2,J,K)*HTMP(2,J,K)+(1.-HTMP(2,J,K))*FTMP(3,J,K)
      F3=FTMP(3,J,K)*HTMP(3,J,K)+(1.-HTMP(3,J,K))*FTMP(2,J,K)
      F4=FTMP(4,J,K)*HTMP(4,J,K)
      A1=HTMP(1,J,K)*HTMP(2,J,K)
      A2=HTMP(2,J,K)*HTMP(3,J,K)
      A3=HTMP(3,J,K)*HTMP(4,J,K)
```



```
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DX1+A2*(F3-F2)/DX2)
      D3=W3*(A2*(F3-F2)/DX2+A3*(F4-F3)/DX3)
      H=HTMP(2,J,K)+HTMP(3,J,K)
      W=WET(INT(H)+1)
      DEL=DX2

      XX=(X-AX(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FY(J,K)=((AA*XX+BB)*XX+CC)*XX+DD
      HY(J,K)=W*H
C     Cubic splane interpolation
C     X1   X2   X3   X4    AX(*)
C     F1   F2   F3   F4     A(*)
C          D2   D3
      ENDDO    ! ..........................  J
      F1=FY(1,K)*HY(1,K)
      F2=FY(2,K)*HY(2,K)+(1.-HY(2,K))*FY(3,K)
      F3=FY(3,K)*HY(3,K)+(1.-HY(3,K))*FY(2,K)
      F4=FY(4,K)*HY(4,K)
      A1=HY(1,K)*HY(2,K)
      A2=HY(2,K)*HY(3,K)
      A3=HY(3,K)*HY(4,K)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DY1+A2*(F3-F2)/DY2)
      D3=W3*(A2*(F3-F2)/DY2+A3*(F4-F3)/DY3)
      H=HY(2,K)+HY(3,K)
      W=WET(INT(H)+1)
      DEL=DY2

      XX=(Y-AY(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FZ(K)=((AA*XX+BB)*XX+CC)*XX+DD
      HZ(K)=W*H
      ENDDO      ! ............................   K
      F1=FZ(1)*HZ(1)
      F2=FZ(2)*HZ(2)+(1.-HZ(2))*FZ(3)
      F3=FZ(3)*HZ(3)+(1.-HZ(3))*FZ(2)
      F4=FZ(4)*HZ(4)
      A1=HZ(1)*HZ(2)
      A2=HZ(2)*HZ(3)
      A3=HZ(3)*HZ(4)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DZ1+A2*(F3-F2)/DZ2)
      D3=W3*(A2*(F3-F2)/DZ2+A3*(F4-F3)/DZ3)
      H=HZ(2)+HZ(3)
      W=WET(INT(H)+1)
      DEL=DZ2
      XX=(Z-P(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
```

```
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FFP=((AA*XX+BB)*XX+CC)*XX+DD
      HHP=W*H

      IF(IXYZ.EQ.1)THEN
      CALL DBGI3D(6,'CUBI3D',FFP,HHP,
     * AX,AY,P,
     * FTMP,HTMP,FY,HY,FZ,HZ)

      ENDIF

      RETURN

      END
C *********************************************
C                                             *
C           FROM    LI1D.f                    *
C                                             *
C *********************************************
C -------------------------------------------
C     3D trajectories:
C
C /users/igor/trj/scarb/claudio/tglob3dfor.f
C
C      27.09.98
Cf77 tglob3dfor.f rec_trj.f -o tglob3dfor.exe
C -------------------------------------------
C23456789 123456789 123456789 123456789 12345
C    |                                       |
C    6                                      45
C|||||||||||||||||||||||||||||||||||||||||||||
C& LI1D
      SUBROUTINE LI1D(FZ,HZ,ZZ,KMAX,ZP,FP,HP)
C
C     LINEAR INTERPOLATION WITH MASK (0./1.)
C     INPUT:
C     FZ(KMAX) is function y=f(x)
C     HZ(KMAX) is H mask hmask=h(x)
C     ZZ(KMAX) is argument x
C     OUTPUT:
C     ZP
C     FP
C     HP
C                    i    i+1
C     x----x----x----x----x----x----x----x
C     1                                KMAX
C
      DIMENSION FZ(KMAX),HZ(KMAX),ZZ(KMAX)
      DIMENSION WET(3)
      DATA WET/0., 1., 0.5/

      Z=ZP

      CALL LOCATE(ZZ,KMAX,Z,KP)

      IF(KP.EQ.0)THEN
      KP=1
      Z=ZZ(1)
      ENDIF
      IF(KP.EQ.KMAX)THEN
      KP=KMAX-1
      Z=ZZ(KMAX)
      ENDIF

      Z1=ZZ(KP)
      Z2=ZZ(KP+1)
      F1=FZ(KP)*HZ(KP)
      F2=FZ(KP+1)*HZ(KP+1)
```



```
      DZF=HZ(KP)*HZ(KP+1)*(F2-F1)/(Z2-Z1)
      H=HZ(KP)+HZ(KP+1)

      IF(H.LE.2)THEN
      W=WET(INT(H)+1)
      ELSE
      W=0.
      ENDIF

      FP=W*(F1+F2+(2.*Z-Z1-Z2)*DZF)
      HP=W*H

      RETURN

      END
C& PROFIL
      SUBROUTINE PROFIL(KP,FF,ZZ,HTM,ZPR,FPR,
     *LEVS,LEV)
C
C     Program computes profile of FF
C     in a grig point.
C     FPR(LEVS)=
C
      DIMENSION ZPR(LEVS),FPR(LEVS)

      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     *          IMJM=IM*JM-JM/2)
      PARAMETER (LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (KMAX=LM)
      PARAMETER (KTOP=KMAX)
      PARAMETER (KMIN=KMAX)
      DIMENSION FF(IMJM,KMAX),HTM(IMJM,KMAX),
     *          ZZ(IMJM,KMAX+1)
                 C O M M O N /MAPOT/
     & CMAPO1(67),SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     $,CMAPO2(8)
C
      DIMENSION FZ(KMAX),HZ(KMAX),PP(KMAX),
     *          ZZ1(KMAX+1),PP1(KMAX+1)

      DO K=1,KMAX
      FZ(K)=FF(KP,K)
      HZ(K)=HTM(KP,K)
      PP(K)=AETA(KMIN+K)
      ZZ1(KMAX-K+2)=ZZ(KP,K)
      PP1(KMAX-K+2)=ETA(KMIN+K)
      ENDDO
      PP1(1)=ETA(LP1)
      ZZ1(1)=ZZ(KP,KMAX+1)
C ||||||||||||||||||||||||||||||
      KK=0
      DO K=1,LEVS
      Z2=ZPR(K)

      IF(Z2.LT.ZZ1(1))Z2=ZZ1(1)
      IF(Z2.GT.ZZ1(KMAX+1))Z2=ZZ1(KMAX+1)

      CALL LINI1D(ZZ1,PP1,KMAX+1,Z2,PLEV,
     *   ILEV)

      CALL LI1D(FZ,HZ,PP,KMAX,PLEV,FP,HP)

      IF(HP.EQ.1.)THEN
      KK=KK+1
      FPR(KK)=FP
      ENDIF
```

```
      ENDDO

      LEV=KK

      RETURN
      END
C& PROFLS
      SUBROUTINE PROFLS(NU,NAME,IP,JP,
     *         FF,ZZ,HTM)
C
C     Program computes profiles
C     from 0 to 3500 m
C
      CHARACTER*(*) NAME
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     *          IMJM=IM*JM-JM/2)
      PARAMETER (LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (KMAX=LM)
      PARAMETER (KTOP=KMAX)
      PARAMETER (KMIN=KMAX)
      DIMENSION FF(IMJM,KMAX),HTM(IMJM,KMAX),
     * ZZ(IMJM,KMAX+1)
       C O M M O N /MAPOT/
     & CMAPO1(67),SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     $,CMAPO2(8)
C
      PARAMETER (LEVS=21)
      DIMENSION ZPR(LEVS),FPR(LEVS)
      DZ=3500./FLOAT(LEVS-1)
      DO K=1,LEVS
      ZPR(K)=(K-1)*DZ
      ENDDO
      KP=INDEXH(IP,JP,IM)

      CALL PROFIL(KP,FF,ZZ,HTM,
     * ZPR,FPR,LEVS,LEV)
      X0= 0.0
      X1=10.0
      DX= 1.0
      MARK=3
      LINE=1
      LITH=3
      CALL GRAPHF(NU,
     *  NAME,
     * 'Excess of concentration',
     * ' Height (m)',
     * ' ',
     * MARK,LINE,LITH,
     *  FPR,ZPR(LEVS-LEV+1),LEV,
     *  X0,X1,DX,
     *  0.,3500.,500.)
      RETURN
      END
C& GRAPHF
      SUBROUTINE GRAPHF(NU,TITL,XLAB,
     *                 YLAB,TEXT,
     *                 MARK,LINE,LITH,
     *                 X,Y,M,
     *                 X0,X1,DX,Y0,Y1,DY)
      CHARACTER*(*) TITL,TEXT,XLAB,YLAB
      DIMENSION X(M),Y(M)
C
C     EXTENDED NEW VERSION OF GRAPH
C     SUBROUTINE
C     31/08/98
C
```



```
      WRITE(NU,'(''#F'')')
      L=LEN(TITL)
      WRITE(NU,101)(TITL(I:I),I=1,L)
      L=LEN(XLAB)
      WRITE(NU,101)(XLAB(I:I),I=1,L)
      L=LEN(YLAB)
      WRITE(NU,101)(YLAB(I:I),I=1,L)
      L=LEN(TEXT)
      WRITE(NU,102)(TEXT(I:I),I=1,L)

      MARK1=MARK
      LINE1=LINE
      LITH1=LITH
      IF(MARK.GE.100)THEN
      MARK1=1
      ENDIF
      IF(LINE.GE.100)THEN
      LINE1=1
      ENDIF
      IF(LITH.GE.7)THEN
      LITH1=6
      ENDIF

      WRITE(NU,'(3(1X,I2))')MARK1,
     *                 LINE1,LITH1
      WRITE(NU,*)X0,X1,Y0,Y1
      WRITE(NU,*)X0,DX,Y0,DY
      DO J=1,M
      WRITE(NU,*)X(J),Y(J)
      ENDDO
      WRITE(NU,*)X(M),Y(M)
      WRITE(NU,'(''end'')')
      RETURN
100   FORMAT(A40)
101   FORMAT(80A1)
102   FORMAT(1X,79A1)

      END
C& GRAPPF
      SUBROUTINE GRAPPF(NU,TEXT,
     *    MARK,LINE,LITH,X,Y,M)
      DIMENSION X(M),Y(M)
      CHARACTER*(*) TEXT
C
C     EXTENDED NEW VERSION OF GRAPH
C     SUBROUTINE
C     31/08/98
C
      BACKSPACE(NU)
      MARK1=MARK
      LINE1=LINE
      LITH1=LITH
      IF(MARK.GE.100)THEN
      MARK1=1
      ENDIF
      IF(LINE.GE.100)THEN
      LINE1=1
      ENDIF
      IF(LITH.GE.7)THEN
      LITH1=6
      ENDIF
      L=LEN(TEXT)
      IF(L.GT.20)L=20
      WRITE(NU,100)MARK1,LINE1,LITH1,
     * (TEXT(I:I),I=1,L)
      DO J=1,M
      WRITE(NU,*)X(J),Y(J)
      ENDDO
      WRITE(NU,*)X(M),Y(M)
```

```
      WRITE(NU,'(''end'')')
      RETURN
100   FORMAT('new',3(1X,I2),1X,20A1)
      END
C& GRAPHC
      SUBROUTINE GRAPHC(NU,TITL,
     * TEXT,XLAB,X,Y,M,X0,X1,DX,
     *                  Y0,Y1,DY,CON)
      CHARACTER*(*) TITL,TEXT,XLAB
      DIMENSION X(M),Y(M)
C
      WRITE(NU,'(''#G'')')
      WRITE(NU,100)TITL
      WRITE(NU,100)XLAB
      WRITE(NU,'('' 2'',A20)')TEXT
      WRITE(NU,*)X0,X1,Y0,Y1
      WRITE(NU,*)X0,DX,Y0,DY
      IC=0
      DO J=1,M
      W=Y(J)
      IF(W.EQ.CON.AND.IC.EQ.0)THEN
      WRITE(NU,'(''new  2'')')
      IC=1
      ELSE
      IF(W.NE.CON)THEN
      WRITE(NU,*)X(J),Y(J)
      IC=0
      ENDIF
      ENDIF
      ENDDO
      WRITE(NU,*)X(M),Y(M)
      WRITE(NU,'(''end'')')
      RETURN
100   FORMAT(A20)
      END
C& GRAPAC
      SUBROUTINE GRAPAC(NU,TEXT,X,Y,M,CON)
      DIMENSION X(M),Y(M)
      CHARACTER*(*) TEXT
C
      BACKSPACE(NU)
      WRITE(NU,100)' 2',TEXT
      IC=0
      DO J=1,M
      W=Y(J)
      IF(W.EQ.CON.AND.IC.EQ.0)THEN
      WRITE(NU,'(''new  2'')')
      IC=1
      ELSE
      IF(W.NE.CON)THEN
      WRITE(NU,*)X(J),Y(J)
      IC=0
      ENDIF
      ENDIF
      ENDDO

      WRITE(NU,*)X(M),Y(M)
      WRITE(NU,'(''end'')')
      RETURN
100   FORMAT('new',A20)
      END
C *********************************************
C                                             *
C           FROM    WROUTP.f                  *
C                                             *
C *********************************************
C& WRHIST
      SUBROUTINE WRHIST(NU,IREC,QQ,ZZ,AU,AV,
     * HTM,HZM,VTM,PLEV1,PLEV2,BB,BU,BV,KMAX,CON)
```



```
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (IMAX=51, JMAX=51)
      DIMENSION HTM(IMJM,KMAX),HZM(IMJM,KMAX+1),
     * VTM(IMJM,KMAX)
      DIMENSION QQ(IMJM,KMAX),ZZ(IMJM,KMAX+1),
     * AU(IMJM,KMAX),AV(IMJM,KMAX),
     * BB(IMAX,JMAX,6),
     * BU(IMAX,JMAX,6),BV(IMAX,JMAX,6),
     * PLEV1(KMAX),PLEV2(KMAX+1)
C ---------------------------------------------
      PARAMETER (LEVS = 6)
      PARAMETER (NEL3 = 3)
      PARAMETER (NEL2 = 1)
      PARAMETER (NSTEP = 9)
C ----------------------------------------------
C      0   12  24  36  48  60  72  84  96  hours
C      x---o---x---o---x---o---x---o---x
C      1   2   3   4   5   6   7   8   9   step
C ----------------------------------------------
      PARAMETER (MAXREC=(LEVS*NEL3+NEL2)*NSTEP)
C ---------------------------------------------
      IREST=MOD(IREC,MAXREC)
      IF(IREST.EQ.0)THEN

      WRITE(66,*)'FIRST HISTORY: IREC=',IREC,' MAXREC=',MAXREC,' #/'

      IREC=0
      ELSE
      IREC=IREST
      ENDIF

      WRITE(66,*)'START HISTORY RECORD=',IREC,' #/'

      CALL WROUT(NU,IREC,QQ,HTM,HZM,
     * PLEV1,KMAX,BB,CON)
      CALL EXTRAC(BB(1,1,3),IMAX*JMAX,
     * AMEAN,AMIN,AMAX,-999.0,NCON)
      WRITE(0,*)'QQ(1,1,3)=',AMEAN,AMIN,AMAX,-999.0,NCON
      WRITE(0,*)'FINISH WROUT(...,QQ'

      CALL WROUT(NU,IREC,ZZ,HZM,HZM,
     * PLEV2,KMAX+1,BB,CON)
      CALL EXTRAC(BB(1,1,3),IMAX*JMAX,
     * AMEAN,AMIN,AMAX,-999.0,NCON)
      WRITE(0,*)'ZZ(1,1,3)=',AMEAN,AMIN,AMAX,-999.0,NCON

      WRITE(0,*)'FINISH WROUT(...,ZZ'

      CALL WROUV(NU,IREC,AU,AV,VTM,HZM,PLEV1,KMAX,
     *              BU,BV,CON)

      CALL EXTRAC(BU(1,1,3),IMAX*JMAX,
     * AMEAN,AMIN,AMAX,-999.0,NCON)
      WRITE(0,*)'UU(1,1,3)=',AMEAN,AMIN,AMAX,-999.0,NCON
      CALL EXTRAC(BV(1,1,3),IMAX*JMAX,
     * AMEAN,AMIN,AMAX,-999.0,NCON)
      WRITE(0,*)'VV(1,1,3)=',AMEAN,AMIN,AMAX,-999.0,NCON
      WRITE(0,*)'FINISH WROUV(...,AU,AV'

      WRITE(66,*)'FINISH HISTORY RECORD=',IREC,' #/'

      RETURN
      END
C& WROUT
      SUBROUTINE WROUT(NU,IREC,AA,HTM,HZM,
     * PLEV,KMAX,BB,CON)
```

```
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AA(IMJM,KMAX),HTM(IMJM,KMAX),
     * HZM(IMJM),BB(IMAX,JMAX,6),PLEV(KMAX)
      DIMENSION PE(LM),WA(LM), MSK(LM),
     *          PB(6),ZWA(LP1),HMSK(LM)
      COMMON/ETAGRD/
     * XB,DEX,YB,DEY,
     * COSETA(JM),XETA(IMT),YETA(JM),
     * ITMP(NPOINT),JTMP(NPOINT),KTMP(NPOINT)
C-----------------------------------------------
       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,
     & DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,U(IMJM,LM),V(IMJM,LM),
     & T(IMJM,LM),Q(IMJM,LM)
C----------------------------------------------
      DATA PB/ 92500., 85000., 70000.,
     *         50000., 30000., 20000./
CX      DATA PB/100000., 92500., 85000.,
CX     *         80000., 72500., 66000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG   !   in radian
       DLAM=DPHD*RADDEG   !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
       DO I=1,IMAX
       XI=-80.0+(I-1)*DLON
       GLON=XI*RADDEG

       CALL ToETA(GLON,GLAT,
     * ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

       IF(
     * XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
```



```
     *                 .AND.
     * YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN

       CALL LI2D(XE,YE,PD,HZM,'H',PDPNT,HPDPNT)
       CALL LI2D(XE,YE,RES,HZM,'H',RESPNT,HREPNT)

C        CALL AV25P(XE,YE,Q,H,IMJM,CON,QP)

       IF(HPDPNT.EQ.0.0.OR.HREPNT.EQ.0.0)THEN
       C1=1.
       ELSE
       C1=1./(PDPNT*RESPNT)
       ENDIF

          DO K=1,KMAX
C
C      X,Y interpolation/averaged
C
      CALL LI2D(XE,YE,AA(1,K),HTM(1,K),'H',WA(K),HMSK(K))
          ENDDO

       DO K=1,6
       ETAL=C1*(PB(K)-PT)
       CALL LI1D(WA,HMSK,PLEV,KMAX,ETAL,AETAL,HETAL)

       IF(HETAL.EQ.1.)THEN
       BB(I,J,K)=AETAL
       ELSE
       BB(I,J,K)=CON
       ENDIF

       ENDDO

       ELSE
C
C      OUT MODEL AREA
C
       DO K=1,6
       BB(I,J,K)=CON
       ENDDO

       ENDIF


       ENDDO     ! I loop
       ENDDO     ! J loop
C
C     FILTR
C
      DO K=1,6
      DO J=1,JMAX
      DO I=1,IMAX
      IF(ABS(BB(I,J,K)).LT.1.E-20)
     * BB(I,J,K)=0.
      ENDDO
      ENDDO
      ENDDO

      DO K=1,6
      IREC=IREC+1
      WRITE(NU,rec=IREC)
     * ((BB(I,J,K),I=1,IMAX),J=1,JMAX)
      ENDDO
       RETURN
       END
C& WROUV
      SUBROUTINE WROUV(NU,IREC,AU,AV,HTM,HZM,PLEV,KMAX,
     *                                       BU,BV,CON)
```

```
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (JM2=JM/2+1,IMT=IM*2-1)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=51, JMAX=51)
C
      DIMENSION AU(IMJM,KMAX),AV(IMJM,KMAX),
     *    HZM(IMJM),HTM(IMJM,KMAX),PLEV(KMAX),
     *          BU(IMAX,JMAX,6),BV(IMAX,JMAX,6)
      DIMENSION PE(LM), WA1(LM), WA2(LM), MSK(LM),
     *          PB(6),HMSK1(LM),HMSK2(LM)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMT),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
C----------------------------------------------------------
                              C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,U(IMJM,LM),V(IMJM,LM),
     & T(IMJM,LM),Q(IMJM,LM)
C--------------------------------------------
      DATA PB/ 92500., 85000., 70000.,
     *         50000., 30000., 20000./
      DATA ERAD /6370000./
C
      ICL=0
      KMIN=LM-KMAX
C
       RADDEG=3.14159265/180.
       DEG=180./3.14159265
C
       DPHI=DPHD*RADDEG    !   in radian
       DLAM=DPHD*RADDEG    !   in radian

       GLON0=TLM0D*RADDEG
       GLAT0=TPH0D*RADDEG

       DLON=1.0
       DLAT=1.0

       DO J=1,JMAX
       YJ=-40.0+(J-1)*DLAT
       GLAT=YJ*RADDEG
       DO I=1,IMAX
       XI=-80.0+(I-1)*DLON
       GLON=XI*RADDEG

       CALL ToETA(GLON,GLAT,ELON,ELAT,GLON0,GLAT0)

       XE=ELON*DEG
       YE=ELAT*DEG

       IF(
     *   XE.GE.XETA(3).AND.XE.LE.XETA(IMT-2)
     *                .AND.
     *   YE.GE.YETA(3).AND.YE.LE.YETA(JM-2)
     *   )THEN
```



```
       CALL LI2D(XE,YE,PD,HZM,'H',PDPNT,HPDPNT)
       CALL LI2D(XE,YE,RES,HZM,'H',RESPNT,HREPNT)

       IF(HPDPNT.EQ.0.0.OR.HREPNT.EQ.0.0)THEN
       C1=1.
       ELSE
       C1=1./(PDPNT*RESPNT)
       ENDIF

          DO K=1,KMAX
C
C      X,Y interpolation for U and V
C      6 is minimum point number
C
      CALL LI2D(XE,YE,AU(1,K),HTM(1,K),'V',WA1(K),HMSK1(K))
      CALL LI2D(XE,YE,AV(1,K),HTM(1,K),'V',WA2(K),HMSK2(K))
          ENDDO

          DO K=1,KMAX
          UR=WA1(K)
          VR=WA2(K)

          IF(HMSK1(K)*HMSK2(K).NE.0.)THEN
          CALL UVGLOB(ELON,ELAT,GLON0,GLAT0,UR,VR,UG,VG)
          WA1(K)=UG
          WA2(K)=VG
          ELSE
          WA1(K)=CON
          WA2(K)=CON
          ENDIF

          ENDDO

       DO K=1,6
       ETAL=C1*(PB(K)-PT)
       CALL LI1D(WA1,HMSK1,PLEV,KMAX,ETAL,AETAL,HETAL)
       IF(HETAL.EQ.1.)THEN
       BU(I,J,K)=AETAL
       ELSE
       BU(I,J,K)=CON
       ENDIF

       CALL LI1D(WA2,HMSK2,PLEV,KMAX,ETAL,AETAL,HETAL)
       IF(HETAL.EQ.1.)THEN
       BV(I,J,K)=AETAL
       ELSE
       BV(I,J,K)=CON
       ENDIF
       ENDDO

       ELSE
          BU(I,J,1)=CON    !   about 1000 hPa
          BU(I,J,2)=CON    !   about  950 hPa
          BU(I,J,3)=CON    !   about  850 hPa
          BU(I,J,4)=CON    !   about  800 hPa
          BU(I,J,5)=CON    !   about  725 hPa
          BU(I,J,6)=CON    !   about  660 hPa

          BV(I,J,1)=CON    !   about 1000 hPa
          BV(I,J,2)=CON    !   about  950 hPa
          BV(I,J,3)=CON    !   about  850 hPa
          BV(I,J,4)=CON    !   about  800 hPa
          BV(I,J,5)=CON    !   about  725 hPa
          BV(I,J,6)=CON    !   about  660 hPa
       ENDIF


       ENDDO     ! I loop
```

```
        ENDDO     ! J loop
C
C      FILTR
C
       DO K=1,6
       DO J=1,JMAX
       DO I=1,IMAX
       IF(ABS(BU(I,J,K)).LT.1.E-10)BU(I,J,K)=0.
       IF(ABS(BV(I,J,K)).LT.1.E-10)BV(I,J,K)=0.
       ENDDO
       ENDDO
       ENDDO

       DO K=1,6
       IREC=IREC+1
       WRITE(NU,rec=IREC)((BU(I,J,K),I=1,IMAX),J=1,JMAX)
       ENDDO

       DO K=1,6
       IREC=IREC+1
       WRITE(NU,rec=IREC)((BV(I,J,K),I=1,IMAX),J=1,JMAX)
       ENDDO

        RETURN
        END
C& LIN1D
       SUBROUTINE LIN1D(X,Y,N,X1,Y1)
       DIMENSION X(N),Y(N)
       PARAMETER (MW=50)
       DIMENSION XX(MW),YY(MW),DYY(MW,2)
       DX=X(N)-X(1)
       IF(DX.GT.0.)THEN
             IF(X1.LT.X(1))THEN
             Y1=Y(1)
             RETURN
             ENDIF
             IF(X1.GT.X(N))THEN
             Y1=Y(N)
             RETURN
             ENDIF
       DO J=1,N
       XX(J)=X(J)
       YY(J)=Y(J)
       ENDDO
       ELSE
             IF(X1.GT.X(1))THEN
             Y1=Y(1)
             RETURN
             ENDIF
             IF(X1.LT.X(N))THEN
             Y1=Y(N)
             RETURN
             ENDIF
       DO J=1,N
       XX(J)=X(N-J+1)
       YY(J)=Y(N-J+1)
       ENDDO
       ENDIF

       CALL LOCATE(XX,N,X1,I)
       IF(I.EQ.N)THEN
       Y1=YY(I)
       RETURN
       ENDIF
       XX1=XX(I)
       YY1=YY(I)
       XX2=XX(I+1)
       YY2=YY(I+1)
       Y1=YY1+(X1-XX1)*(YY2-YY1)/(XX2-XX1)
```



```
       RETURN
       END
C *********************************************
C                                             *
C            FROM    I2DHV.f                  *
C                                             *
C *********************************************
C& INDEX
C ************************************************
C       Subroutine set for plane interpolation
C       on Eta-Grid with Mask
C ************************************************
C             I N D E X
C
C     No Name           Origin
C     1. LI2D        from LI3D
C     2. CUBI2D      from CUBI3D
C     3. D2HARR      from CUBARR
C     4. D2VARR      from CUBVAR
C
C& LI2D
       SUBROUTINE LI2D(XP,YP,FF,HTM,TYPE,FFP,HHP)
C      -----------------------------------
C      See /users/igor/trj/eta/simul/i3d.f
C      -----------------------------------
       PARAMETER (IM=111,JM=221)
       PARAMETER (IMT=2*IM-1,
      *         IMJM=IM*JM-JM/2)
       PARAMETER (LM=38)
       PARAMETER (KMAX5=LM+5)
       PARAMETER (NPOINT=25)
       COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
C
       DIMENSION AX(4),AY(4)
       DIMENSION FF(IMJM),
      * HTM(IMJM)
       CHARACTER*1 TYPE
       DIMENSION FTMP(4,4),
      * HTMP(4,4), FY(4),HY(4)
       DIMENSION WET(3)
       DATA WET/0., 1., 0.5/
       DATA C1/0.7071068/
       DATA C2/1.414214 /
       DATA AX/-1.414214, 0.,
      *         1.414214, 2.828428/
       DATA AY/-1.414214, 0.,
      *         1.414214, 2.828428/
C
C       WRITE(0,*)'LI2D:',TYPE,' ',XP,YP
C
       IF(TYPE.EQ.'H')THEN
       CALL D2HARR
      *(XP,YP,FF,HTM,
      * FTMP,HTMP,
      * XB,YB,DX,DY)
       GOTO 500
       ENDIF

       IF(TYPE.EQ.'V')THEN
       CALL D2VARR
      *(XP,YP,FF,HTM,
      * FTMP,HTMP,
      * XB,YB,DX,DY)
       GOTO500
       ENDIF

           WRITE(66,*)'ERROR IN LI2D'
           WRITE(66,*)'TYPE=',TYPE
```

```
          STOP

500   CONTINUE

      XLOC=(XP-XB)/DX
      YLOC=(YP-YB)/DY

      X1=0.
      X2=C2
      X= C1*(XLOC+YLOC)
      Y1=0.
      Y2=C2
      Y=-C1*(XLOC-YLOC)

      DO J=2,3
      F1=FTMP(2,J)*HTMP(2,J)
      F2=FTMP(3,J)*HTMP(3,J)
      DXF=HTMP(2,J)*HTMP(3,J)
     *        *(F2-F1)/(X2-X1)
      H=HTMP(2,J)+HTMP(3,J)
      W=WET(INT(H)+1)
      FY(J)=W*(F1+F2+(2.*X-X1-X2)
     *                      *DXF)
      HY(J)=W*H
      ENDDO      !   J
      F1=FY(2)*HY(2)
      F2=FY(3)*HY(3)
      DYF=HY(2)*HY(3)*(F2-F1)
     *                      /(Y2-Y1)
      H=HY(2)+HY(3)
      W=WET(INT(H)+1)
      FFP=W*(F1+F2+(2.*Y-Y1-Y2)*DYF)
      HHP=W*H

      RETURN

      END
C& CUBI2D
      SUBROUTINE CUBI2D(XP,YP,
     * FF,HTM,TYPE,FFP,HHP)

      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,IMJM=IM*JM-JM/2)
      PARAMETER (NPOINT=25)
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
      COMMON/DBGXYZ/IXYZ

      DIMENSION AX(4),AY(4)
      DIMENSION FF(IMJM),HTM(IMJM)
      CHARACTER*1 TYPE
      DIMENSION FTMP(4,4),HTMP(4,4)
      DIMENSION FY(4),HY(4)
      DIMENSION WET(3)
      DATA WET/0., 1., 0.5/
      DATA C1/0.7071068/
      DATA C2/1.414214 /
      DATA AX/-1.414214, 0.,
     *         1.414214, 2.828428/
      DATA AY/-1.414214, 0.,
     *         1.414214, 2.828428/
C
      IF(TYPE.EQ.'H')THEN
      CALL D2HARR
     *(XP,YP,FF,HTM,
     * FTMP,HTMP,
     * XB,YB,DX,DY)
      GOTO500
      ENDIF
```



```
      IF(TYPE.EQ.'V')THEN
      CALL D2VARR
     *(XP,YP,FF,HTM,
     * FTMP,HTMP,
     * XB,YB,DX,DY)
      GOTO500
      ENDIF
          WRITE(66,*)'ERROR IN LI2D'
          WRITE(66,*)'TYPE=',TYPE
          STOP
500   CONTINUE

      XLOC=(XP-XB)/DX
      YLOC=(YP-YB)/DY

      DX1=AX(2)-AX(1)
      DX2=AX(3)-AX(2)
      DX3=AX(4)-AX(3)
      DY1=AY(2)-AY(1)
      DY2=AY(3)-AY(2)
      DY3=AY(4)-AY(3)

      X= C1*(XLOC+YLOC)
      Y=-C1*(XLOC-YLOC)
C
C        F1      F2      F3      F4
C     4  x------x------x------x
C        |      |      |      |
C        |  A1  |  A2  |  A3  |
C     3  x------x------x------x
C        |      W2     W3     |
C        |  DX1 |  DX2 |  DX3 |
C     2  x------x------x------x
C        |      |      |      |
C        |      |      |      |
C     1  x------x------x------x
C        1      2      3      4
C
C


      DO J=1,4
      F1=FTMP(1,J)*HTMP(1,J)
      F2=FTMP(2,J)*HTMP(2,J)+(1.-HTMP(2,J))*FTMP(3,J)
      F3=FTMP(3,J)*HTMP(3,J)+(1.-HTMP(3,J))*FTMP(2,J)
      F4=FTMP(4,J)*HTMP(4,J)
      A1=HTMP(1,J)*HTMP(2,J)
      A2=HTMP(2,J)*HTMP(3,J)
      A3=HTMP(3,J)*HTMP(4,J)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DX1+A2*(F3-F2)/DX2)
      D3=W3*(A2*(F3-F2)/DX2+A3*(F4-F3)/DX3)
      H=HTMP(2,J)+HTMP(3,J)
      W=WET(INT(H)+1)
      DEL=DX2

      XX=(X-AX(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FY(J)=((AA*XX+BB)*XX+CC)*XX+DD
      HY(J)=W*H
C     Cubic splane interpolation
C     X1   X2   X3   X4    AX(*)
C     F1   F2   F3   F4     A(*)
```

```
C          D2   D3
      ENDDO    ! ..........................  J
      F1=FY(1)*HY(1)
      F2=FY(2)*HY(2)+(1.-HY(2))*FY(3)
      F3=FY(3)*HY(3)+(1.-HY(3))*FY(2)
      F4=FY(4)*HY(4)
      A1=HY(1)*HY(2)
      A2=HY(2)*HY(3)
      A3=HY(3)*HY(4)
      W2=WET(INT(A1+A2)+1)
      W3=WET(INT(A2+A3)+1)
      D2=W2*(A1*(F2-F1)/DY1+A2*(F3-F2)/DY2)
      D3=W3*(A2*(F3-F2)/DY2+A3*(F4-F3)/DY3)
      H=HY(2)+HY(3)
      W=WET(INT(H)+1)
      DEL=DY2

      XX=(Y-AY(2))

      DD=F2
      CC=D2
      FF1=F3-F2-D2*DEL
      FF2=D3-D2
      AA=(DEL*FF2-2.*FF1)/DEL**3
      BB=(FF1-AA*DEL**3)/DEL**2
      FFP=((AA*XX+BB)*XX+CC)*XX+DD
      HHP=W*H

      RETURN

      END
C& D2HARR
      SUBROUTINE D2HARR
     *(X,Y,FF,HTM,FTMP,HTMP,
     * XBA,YBA,DXB,DYB)
C
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
CX    ///////////////////////////
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
CX    ///////////////////////////
      DIMENSION FF(IMJM),HTM(IMJM)
      DIMENSION FTMP(16),HTMP(16)
C
      COMMON/ETAGRD/
     * XB,DEX,YB,DEY,
     * COSETA(JM),XETA(IMT),YETA(JM),
     * ITMP(NPOINT),JTMP(NPOINT),KTMP(NPOINT)
C
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
      ITYPE=0

      DX=(X-XB)/DEX
      DY=(Y-YB)/DEY
C
C     Base index
C
      I1=INT(0.5*DX)+1
      J1=INT(0.5*DY)+1

      I=INT(DX)+1
      J=INT(DY)+1
C
C     Local coordinate
C
```



```
      X0=XB+(I1-1)*DEX*2.
      Y0=YB+(J1-1)*DEY*2.

      XBQ=X0
      YBQ=Y0
C
C     Base Point Index (address)
C
      K0=(J1-1)*IMT+I1
C
      I1=2*I1-1
      J1=2*J1-1

      IF(I1.EQ.I)THEN
         IF(J1.EQ.J)THEN
C                      I case
            XP=(X-X0)/DEX
            YP=(Y-Y0)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0
            XBA=X0
            YBA=Y0
            ITYPE=10
            ELSE
            KADDR=K0-IM+1
            XBA=X0+DEX
            YBA=Y0-DEY
            ITYPE=11
            ENDIF
         ELSE
C                    III case
            XP=(X-X0)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(XP+YP.LT.1.)THEN
            KADDR=K0
            XBA=X0
            YBA=Y0
            ITYPE=31
            ELSE
            KADDR=K0+IM
            XBA=X0+DEX
            YBA=Y0+DEY
            ITYPE=30
            ENDIF
         ENDIF
       ELSE
         IF(J1.EQ.J)THEN
C                     II case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0)/DEY
            IF(XP+YP.GE.1.)THEN
            KADDR=K0+1
            XBA=X0+2.*DEX
            YBA=Y0
            ITYPE=20
            ELSE
            KADDR=K0-IM+1
            XBA=X0+DEX
            YBA=Y0-DEY
            ITYPE=21
            ENDIF
         ELSE
C                     IV case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0+IM
            XBA=X0+DEX
            YBA=Y0+DEY
```

```
            ITYPE=40
            ELSE
            KADDR=K0+1
            XBA=X0+2.*DEX
            YBA=Y0
            ITYPE=41
            ENDIF

         ENDIF
      ENDIF
C

      DO J=1,16
      J1=IOAC(J)
      K=KADDR+IOA(J1)
      FTMP(J)=FF(K)
      HTMP(J)=HTM(K)
      ENDDO

      DXB=DEX
      DYB=DEY

      RETURN
      END
C& D2VARR
      SUBROUTINE D2VARR
     *(X,Y,FF,HTM,
     * FTMP,HTMP,
     * XBA,YBA,DXB,DYB)
      PARAMETER (IM=111,JM=221)
      PARAMETER (IMT=2*IM-1,
     * IMJM=IM*JM-JM/2)
      PARAMETER (LM=38)
      PARAMETER (KMAX5=LM+5)
      PARAMETER (NPOINT=25)
CX    ///////////////////////////
      COMMON/CLI3D/IXJ,ITYPE,XBQ,YBQ
CX    ///////////////////////////
      DIMENSION FF(IMJM), HTM(IMJM)
      DIMENSION FTMP(16), HTMP(16)
C
      COMMON/ETAGRD/
     * XB,DEX,YB,DEY,
     * COSETA(JM),XETA(IMT),YETA(JM),
     * ITMP(NPOINT),JTMP(NPOINT),KTMP(NPOINT)
C
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
      ITYPE=0

      DX=(X-XB)/DEX
      DY=(Y-YB)/DEY
C
C     Base index
C
      I1=INT(0.5*DX)+1
      J1=INT(0.5*DY)+1

      I=INT(DX)+1
      J=INT(DY)+1
C
C     Local coordinate
C
      X0=XB+(I1-1)*DEX*2.
      Y0=YB+(J1-1)*DEY*2.

      XBQ=X0
      YBQ=Y0
C
```



```
C     Base Point Index (address)
C
      K0=(J1-1)*IMT+I1
C
      I1=2*I1-1
      J1=2*J1-1

      IF(I1.EQ.I)THEN
         IF(J1.EQ.J)THEN
C                      I case
            XP=(X-X0)/DEX
            YP=(Y-Y0)/DEY
            IF(YP+YP.GE.1.)THEN
            KADDR=K0
            XBA=X0+DEX
            YBA=Y0
            ITYPE=10
            ELSE
            KADDR=K0-IM
            XBA=X0
            YBA=Y0-DEY
            ITYPE=11
            ENDIF
         ELSE
C                    III case
            XP=(X-X0)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0+IM-1
            XBA=X0
            YBA=Y0+DEY
            ITYPE=31
            ELSE
            KADDR=K0
            XBA=X0+DEX
            YBA=Y0
            ITYPE=10
            ENDIF
         ENDIF
       ELSE
         IF(J1.EQ.J)THEN
C                     II case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0)/DEY
            IF(YP.GE.XP)THEN
            KADDR=K0
            XBA=X0+DEX
            YBA=Y0
            ITYPE=10
            ELSE
            KADDR=K0-IM+1
            XBA=X0+2.*DEX
            YBA=Y0-DEY
            ITYPE=12
            ENDIF
         ELSE
C                     IV case
            XP=(X-X0-DEX)/DEX
            YP=(Y-Y0-DEY)/DEY
            IF(YP+YP.GE.1.)THEN
            KADDR=K0+IM
            XBA=X0+2.*DEX
            YBA=Y0+DEY
            ITYPE=13
            ELSE
            KADDR=K0
            XBA=X0+DEX
            YBA=Y0
            ITYPE=10
```

```
            ENDIF

         ENDIF
      ENDIF
C
      DO J=1,16
      J1=IOAC(J)
      K=KADDR+IOA(J1)
      FTMP(J)=FF(K)
      HTMP(J)=HTM(K)
      ENDDO

      DXB=DEX
      DYB=DEY

      RETURN
      END
C& DBGI2D
      SUBROUTINE DBGI2D(NU,NAME,FFP,HHP,
     * AX,AY,
     * FTMP,HTMP,FY,HY)
      CHARACTER*(*) NAME
      DIMENSION AX(4),AY(4)
      DIMENSION FTMP(4,4),HTMP(4,4)
      DIMENSION FY(4),HY(4)
      WRITE(NU,*)NAME,' DEBUGING'
      WRITE(NU,*)'RESULT=',FFP,HHP

      DO J=4,1,-1
      WRITE(NU,100)J,
     * (FTMP(I,J),I=1,4),
     * (HTMP(I,J),I=1,4)
      ENDDO    ! J
      WRITE(NU,*)' '


      WRITE(NU,*)' '
      WRITE(NU,*)'  FY(J)    HY(J)'
      DO J=4,1,-1
      WRITE(NU,101)J,
     * FY(J),
     * HY(J)
      ENDDO

      WRITE(NU,*)' '

      RETURN
100   FORMAT(I3,1X,4(F11.4,1X),3X,4(F4.1,1X))
101   FORMAT(I3,1X,4(1X,F10.6),2X,4(1X,F6.2))
      END
C& VEGNUM
      SUBROUTINE VEGNUM(SOVEG,NUMVEG)
C
C     Subroutines transfors vegetation simbol SOVEG
C     to vegetation number NUMVEG.
C
      CHARACTER*1 COD(14)
      CHARACTER*1 SOVEG

      DATA COD/'0','1','2','3','4',
     *         '5','6','7','8','9',
     *         'A','B','C','D'/
      NUM=-1
      DO K=1,14
      IF(SOVEG.EQ.COD(K))NUM=K-1
      ENDDO

      NUMVEG=NUM
      RETURN
```



```
      END
C& VEGTYP
      SUBROUTINE VEGTYP(X,Y,XBOX,YBOX,VEGS,IMAX,JMAX,NUMVEG)
      DIMENSION XBOX(IMAX+1),YBOX(JMAX+1)
      DIMENSION VEGS(IMAX,JMAX)
      CHARACTER*1 VEGS, S1

      X1=X
CX      IF(X.GT.180.)X1=X-180.
CX      IF(X.LT.-180.)X1=X+180.
      CALL LOCATE(XBOX,IMAX+1,X,I)
      CALL LOCATE(YBOX,JMAX+1,Y,J)
      S1=VEGS(I,J)

      CALL VEGNUM(S1,NUM)
      NUMVEG=NUM

      RETURN
      END
C& VEGETY
      SUBROUTINE VEGETY
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (NPOINT=25)
      PARAMETER (IMAX=360)
      PARAMETER (JMAX=180)
C
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/ETAREA/CAREA(IMJM)
C
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/ETAVEG/IHVEG(IMJM)
      DIMENSION XBOX(IMAX+1),YBOX(JMAX+1)
      DIMENSION VEGS(IMAX,JMAX)
      CHARACTER*1 VEGS
      CHARACTER*360 S360
      CHARACTER*30 TEXT
      DIMENSION DIST(14), TAREA(14)

      PARAMETER (NFIL = 10)
      COMMON/DIRICT/DIRTAB(NFIL),IDIR(NFIL)
      CHARACTER*80   DIRTAB

      I1=IDIR(4)
      OPEN(50,FILE=DIRTAB(4)(1:I1)//'box.dat',
     * STATUS='OLD')

CX      OPEN(50,FILE=
CX     *'/pesqout/sdg04/igor/eta/40km/run/etafcst/box.dat',
CX     * STATUS='OLD')

      WRITE(66,*)'=///////////////////////////=' 
      WRITE(66,*)'    VEGETATION TYPES'
      WRITE(66,*)'=///////////////////////////='
      WRITE(66,*)'      From'

      WRITE(66,*)
     *DIRTAB(4)(1:I1)//'box.dat'
      WRITE(66,*)
     *DIRTAB(4)(1:I1)//'vegs.dat'

CX      WRITE(66,*)
CX     *'/pesqout/sdg04/igor/eta/40km/run/etafcst/box.dat'
CX      WRITE(66,*)
CX     *'/pesqout/sdg04/igor/eta/40km/run/etafcst/vegs.dat'
```

```
      READ(50,'(A30)')TEXT
      WRITE(66,*)TEXT
      READ(50,*)NBX
      DO I=1,NBX
      READ(50,*)IX,XBOX(I)
      ENDDO
      READ(50,'(A30)')TEXT
      WRITE(66,*)TEXT
      READ(50,*)NBY
      DO I=1,NBY
      READ(50,*)IY,YBOX(I)
      ENDDO
      WRITE(66,*)'NBX=',NBX,' NBY=',NBY
      CLOSE(50)

      OPEN(50,FILE=
     *DIRTAB(4)(1:I1)//'vegs.dat',
     * STATUS='OLD')

CX      OPEN(50,FILE=
CX     *'/pesqout/sdg04/igor/eta/40km/run/etafcst/vegs.dat',
CX     * STATUS='OLD')

      READ(50,'(A30)')TEXT
      READ(50,*)NBX,NBY
      DO J=1,JMAX
      READ(50,'(A360)')S360
      DO I=1,360
      VEGS(I,J)=S360(I:I)
      ENDDO
      ENDDO
      CLOSE(50)

      WRITE(66,*)' VEGS'
      DO J=1,20
      WRITE(66,'(30A1)')(VEGS(I,J),I=1,30)
      ENDDO


      RMEAN=0.
      XMEAN=0.
      YMEAN=0.
      NMEAN=0


      DO J=1,IMJM
      IHVEG(K)=0
      ENDDO

      DO IP=1,IPN
      K=INP(IP)
      I=IINP(IP)
      J=JINP(IP)
      CALL GCOORD(I,J,GLON,GLAT)
      CALL VEGTYP(GLON,GLAT,XBOX,YBOX,VEGS,IMAX,JMAX,NUMVEG)
      IHVEG(K)=NUMVEG

      RMEAN=RMEAN+FLOAT(NUMVEG)
      XMEAN=XMEAN+GLON
      YMEAN=YMEAN+GLAT
      NMEAN=NMEAN+1

      ENDDO

      WRITE(66,*)'MEAN TYPE=',RMEAN/FLOAT(NMEAN)
      WRITE(66,*)'MEAN   X=',XMEAN/FLOAT(NMEAN)
      WRITE(66,*)'MEAN   Y=',YMEAN/FLOAT(NMEAN)
      WRITE(66,*)'RMEAN  =',RMEAN
```



```
      WRITE(66,*)'NMEAN  =',NMEAN

      DO I=1,14
      DIST(I)=0.
      TAREA(I)=0.
      ENDDO

      NN=0
      DO IP=1,IPN

      K=INP(IP)
      KT=IHVEG(K)+1
      IF(KT.LT.1.OR.KT.GT.14)THEN
      NN=NN+1
      ELSE
      DIST(KT)=DIST(KT)+1.
      TAREA(KT)=TAREA(KT)+CAREA(K)
      ENDIF

      ENDDO

      WRITE(66,*)'    Distribution of vegetation types'
      WRITE(66,*)'     at Eta Grid inner points(in %)'
      WRITE(66,*)'          and area (km**2)'
      DO I=1,14
      DIST(I)=100.*DIST(I)/FLOAT(IPN)
      ENDDO

      DO I=1,14
      WRITE(66,*)I-1,DIST(I),TAREA(I)/1.E6
      ENDDO

      AREA1=0.
      DO IP=1,IPN
      K=INP(IP)
      IF(IHVEG(K).EQ.1)THEN
      AREA1=AREA1+CAREA(K)
      ENDIF
      ENDDO

      WRITE(66,*)'AREA WITH VEGETATION MASK=1'
      WRITE(66,*)'AREA1=',AREA1,' m**2'
      WRITE(66,*)'AREA1=',AREA1/1.E6,' km**2'
      WRITE(66,*)' '
      WRITE(66,*)'Bad points number=',NN,' IPN=',IPN
      WRITE(66,*)'=//////////////////////////////='

CX      STOP

      RETURN
      END
C ***********************************************************
C                      BUFFERED READING
C ***********************************************************

C& RDFRS
      SUBROUTINE RDFRS(NU,ISTEP,IREC,BUFI,FIRE,IMJM)
      DIMENSION BUFI(IMJM,2),FIRE(IMJM)

      IDAY   = (ISTEP-1)/30+1       !             day
      TIME   = (IDAY-1)*24.*60.     !      min    time

      ISTED  = MOD(ISTEP-1,30)+1    !             inner day step
      TIMED  = ISTED*48.            !      min    inner time
      TBASE  = (IDAY-1)*24.*60.     !      min     base time
      TDAY   = TBASE+TIMED          !      min     time

         IF(ISTED.EQ.16)THEN
         IREC=IREC+1
```

```
      WRITE(66,*)'READ NEXT ISTED=',ISTED,' IREC=',IREC
      CALL DIRDBU(NU,'FIRES',IMJM,IREC,BUFI)
      ENDIF

      IF(ISTED.EQ.19)THEN
      IREC=IREC+1
      WRITE(66,*)'READ NEXT ISTED=',ISTED,' IREC=',IREC
      CALL DIRDBU(NU,'FIRES',IMJM,IREC,BUFI)
      ENDIF

      IF(ISTED.EQ.23)THEN
      IREC=IREC+1
      WRITE(66,*)'READ NEXT ISTED=',ISTED,' IREC=',IREC
      CALL DIRDBU(NU,'FIRES',IMJM,IREC,BUFI)
      ENDIF

      IF(ISTED.EQ.27)THEN
      IREC=IREC+1
      WRITE(66,*)'READ NEXT ISTED=',ISTED,' IREC=',IREC
      CALL DIRDBU(NU,'FIRES',IMJM,IREC,BUFI)
      ENDIF

      IF(ISTED.LE.15)THEN
      TFIRE1=TBASE-4.*48
      TFIRE2=TBASE+15.*48.
      NOB1=1
      NOB2=2
      ELSE
         IF(ISTED.LE.18)THEN
         TFIRE1=TBASE+15.*48.
         TFIRE2=TBASE+18.*48.
         NOB1=2
         NOB2=1
         ELSE
            IF(ISTED.LE.22)THEN
            TFIRE1=TBASE+18.*48.
            TFIRE2=TBASE+22.*48.
            NOB1=1
            NOB2=2
            ELSE
               IF(ISTED.LE.26)THEN
               TFIRE1=TBASE+22.*48.
               TFIRE2=TBASE+26.*48.
               NOB1=2
               NOB2=1
               ELSE
               TFIRE1=TBASE+26.*48.
               TFIRE2=TBASE+46.*48.
               NOB1=1
               NOB2=2
               ENDIF
            ENDIF
          ENDIF
       ENDIF

        D1=TFIRE1
        D2=TFIRE2
        TDAY   = TBASE + TIMED

        NOINT=1
        IF(ISTED.EQ.15)NOINT=0
        IF(ISTED.EQ.18)NOINT=0
        IF(ISTED.EQ.22)NOINT=0
        IF(ISTED.EQ.26)NOINT=0

        IF(NOINT.EQ.0)THEN
          DO J=1,IMJM
          FIRE(J)=BUFI(J,NOB2)
          ENDDO
```



```
          ELSE
            DO J=1,IMJM
            F1=BUFI(J,NOB1)
            F2=BUFI(J,NOB2)
            FIRE(J)=F1+(TDAY-D1)*(F2-F1)/(D2-D1)
            ENDDO
          ENDIF

         TIME=TIME/60.
         TIMED=TIMED/60.
         TFIRE1=TFIRE1/60.
         TFIRE2=TFIRE2/60.
         TDAY=TDAY/60.

      IF(ISTEP.EQ.1)THEN
      WRITE(66,*)'ISTEP,IDAY,TIME,TFIRE1,TDAY,TFIRE2,ISTED,NOB1,NOB2'
      ENDIF
      WRITE(66,100)ISTEP,IDAY,TIME,TFIRE1,TDAY,TFIRE2,ISTED,NOB1,NOB2,
     * NOINT
100   FORMAT(I3,1X,I2,2X,4(1X,F6.2),2X,I3,1X,2(1X,I2),' NOINT=',I1)

      WRITE(66,*)
     *'FIRE AREA OF IDAY=',IDAY,' HOURE=',TIME,' IREC=',IREC,' /#'

      RETURN
      END
C& DIRDBU
      SUBROUTINE DIRDBU(NU,NAME,NMAX,IREC,BUF)
C    ///////////////////////////////////////
C     Direct reading from file to buffer
C     Special case: 2 buffers, 1 level, and
C                   1st record into 2nd buffer
C    ///////////////////////////////////////
C    NU      UNIT NUMBER
C    NAME    VARIABLE NAME
C    NMAX    IMAX*JMAX
C    IREC    RECORD NUMBER
C    BUF     BUFFER OF VARIABLE
C    WA      WORK ARREA WA(NMAX)
C    ///////////////////////////////////////
C    Day  number   IREC  1 2 3 4 5 6 7 8 ...
C    Buffer number  NOB  2 1 2 1 2 1 2 1 ...
C    ///////////////////////////////////////
      DIMENSION BUF(NMAX,2)

      IB=MOD(IREC,2)
      IF(IB.EQ.1)THEN
      NOB=2
      ELSE
      NOB=1
      ENDIF

      READ(NU,rec=IREC)(BUF(J,NOB),J=1,NMAX)

      RETURN
      END
C& NULLI
      SUBROUTINE NULLI(A,M)
      DIMENSION A(M)
      DO J=1,M
      A(J)=0.
      ENDDO
      RETURN
      END
C& PARARE
      SUBROUTINE PARARE(ALON,ALAT,I0,J0,GLON0,GLAT0)
C-----------------------------------------------------------
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
```

```
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *       XB,DEX,YB,DEY,
     *       COSETA(JM),XETA(IMAX),YETA(JM),
     *       ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/ETAREA/CAREA(IMJM)
COMMENT
C     ALON,ALAT .................. in degrees
C
C
C     XB,DEX,YB,DEY .............. in degrees
C     GLON0,GLAT0 ................ in radian
C     See VERPRF
C-----------------------------------------------------------
      DATA PI/3.14159265/

      DEG=180./PI
      RAD=PI/180.
      WRITE(66,*)'PARARE SUBROUTINE:'
      WRITE(66,*)'XB, DEX=',XB, DEX
      WRITE(66,*)'YB, DEY=',YB, DEY
      WRITE(66,*)'DEG=',DEG,' RAD=',RAD
      WRITE(66,*)'GLON0,GLAT0=',GLON0,GLAT0

      ALON1=ALON*RAD
      ALAT1=ALAT*RAD
      CALL ToETA(ALON1,ALAT1,ELON,ELAT,GLON0,GLAT0)
      XE=ELON*DEG
      YE=ELAT*DEG
      DXX=XE-XB
      I=INT(DXX/DEX)+1
      IF(MOD(DXX,DEX).EQ.0.0)I=I-1
      DYY=YE-YB
      J=INT(DYY/DEY)+1
      IF(MOD(DYY,DEY).EQ.0.0)J=J-1

      WRITE(66,*)'XE =',XE, ' YE=',YE
      WRITE(66,*)'DXX=',DXX,' DYY=',DYY
CX      K=INDXHC(XE,YE)
CX      CALL INDEXS(K,IMAX,IM,I,J)

      IPJ=I+J
      IF(MOD(IPJ,2).NE.0)I=I-1
      I0=I
      J0=J

      WRITE(66,*)'ALON=',ALON,    ' ALAT=',ALAT
      WRITE(66,*)' XE=',DEG*XE,  ' YE=',DEG*YE
      WRITE(66,*)' I0=',I0,      ' J0=',J0

      K=INDXHC(XE,YE)
      CALL INDEXS(K,IMAX,IM,I1,J1)
      IPJ=I1+J1
      IF(MOD(IPJ,2).NE.0)I1=I1-1
      WRITE(66,*)' I0=',I1,      ' J0=',J1

      RETURN
      END
C& PRFARE
      SUBROUTINE PRFARE(NU,ISTEP,DT,
     * Q,QFLUX,HTM,U,V,VTM,ZZ,PS,IWA,LEN)
C
C  MEAN PROFILES CALCULATION SUBROUTINE FOR AREA
C
C     IWA(LEN)       area addresses
C             1  2   3    4    5   ...
```



```
C     IWA(*)=IP  LEN NINN NWE NSN [NINN],[NWE],[NWE],[NSN],[NSN]
C     IP=0   no
C     IP=1   yes
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)
      PARAMETER (NPRF=10)

      DIMENSION Q(IMJM,KMAX,2),QFLUX(IMJM),
     * ZZ(IMJM,KMAX+1,2), HTM(IMJM,KMAX),
     * U(IMJM,KMAX,2),V(IMJM,KMAX,2),VTM(IMJM,KMAX),
     * PS(IMJM,2),
     * IWA(LEN),
     * PROF(KMAX,NPRF)
C-----------------------------------------------------------
C     PROF(KMAX,1)  -  Mean P    H/m**2
C     PROF(KMAX,2)  -  Mean Z    m
C     PROF(KMAX,3)  -  Mean q    ppm
C     PROF(KMAX,4)  -  Air mass  kg
C     PROF(KMAX,5)  -  Mask
C     PROF(KMAX,6)  -
C     PROF(KMAX,7)  -
C     PROF(KMAX,8)  -
C     PROF(KMAX,9)  -
C     PROF(KMAX,10) -
C-----------------------------------------------------------
       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,UETA(IMJM,LM),VETA(IMJM,LM),
     & TETA(IMJM,LM),QETA(IMJM,LM)
C-----------------------------------------------------------
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *       XB,DEX,YB,DEY,
     *       COSETA(JM),XETA(IMAX),YETA(JM),
     *       ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/ETAREA/CAREA(IMJM)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C-----------------------------------------------------------
      DATA G/9.81/
      DATA PI/3.14159265/
      DATA ERAD /6370000./
C
C     S T A R T
C
      IF(IWA(1).EQ.0)THEN
      WRITE(NU,*)'AREA PARAMETERS ARE EMPTY'
      STOP
      ENDIF
      NINN=IWA(3)
      NWE=IWA(4)
      NSN=IWA(5)

      IBASE=5
      INNER=IBASE
```

```
      IWB=IBASE+NINN
      IEB=IWB+NWE
      ISB=IEB+NWE
      INB=ISB+NSN

      DO L=1,LM

      SMEAN=0.
      PMEAN=0.
      ZMEAN=0.
      TOTM =0.
      AIRMA=0.
      HMSK =0.

      DO J=1,NINN
      K=IWA(INNER+J)
      DP= DETA(L)*PS(K,1)*RES(K)
      AIR=DP/G
      H=HTM(K,L)
      SMEAN=SMEAN+CAREA(K)
      ZMEAN=ZMEAN+0.5*(ZZ(K,L,1)+ZZ(K,L+1,1))*CAREA(K)
      PMEAN=PMEAN + AETA(L)*PS(K,1)*RES(K)*CAREA(K)
      TOTM=TOTM+Q(K,L,1)*AIR*CAREA(K)*H
      AIRMA=AIRMA+AIR*CAREA(K)*H
      HMSK = HMSK + H
      ENDDO

      PROF(L,1)=PMEAN/SMEAN    !  Mean P     H/m**2
      PROF(L,2)=ZMEAN/SMEAN    !  Mean Z     m
      IF(HMSK.NE.0.)THEN
      PROF(L,3)=TOTM/AIRMA     !  Mean q     ppm
      PROF(L,4)=AIRMA          !  Air mass   kg
      PROF(L,5)=1.             !  Mask
C     PROF(KMAX,6)  -
C     PROF(KMAX,7)  -
C     PROF(KMAX,8)  -
C     PROF(KMAX,9)  -
C     PROF(KMAX,10) -
      ELSE
      PROF(L,3)=-999.0
      PROF(L,4)=-999.0
      PROF(L,5)=0.
      ENDIF

      ENDDO

      WRITE(NU,*)'AREA MEAN PROFILES: ISTEP=',ISTEP,' #/'
      DO L=1,KMAX
      WRITE(NU,*)L,
     * PROF(L,1),'(H/m**2)',
     * PROF(L,2),'(m)',
     * PROF(L,3),'(ppm)',
     * PROF(L,4),'(kg)',
     * PROF(L,5),' #/'
      ENDDO

      RETURN
      END
C& BALARE
      SUBROUTINE BALARE(I0,NX,J0,NY,DT,
     * Q,QFLUX,HTM,U,V,VTM,PS,IWA,LEN,COBAL)
C
C     BALANCE CALCULATION SUBROUTINE FOR AREA
C
C     I0,NX,J0,NY    area parameters
C     IWA(LEN)       area addresses
C             1  2   3    4   5   ...
C     IWA(*)=IP  LEN NINN NWE NSN [NINN],[NWE],[NWE],[NSN],[NSN]
C     IP=0   no
```



```
C     IP=1   yes
C     COBAL(10)       Balance components
C     COBAL(1)= TOTM  Total mass for t
C     COBAL(2)= FBOT  Flux from bottom
C     COBAL(3)= FLAT  Lateral boundaries flux
C     COBAL(4)= FSOU  South lateral boundary flux
C     COBAL(5)= FEST  East lateral boundary flux
C     COBAL(6)= FNOR  North lateral boundary flux
C     COBAL(7)= FWES  West lateral boundary flux
C     COBAL(8)=       Area
C
C     COBAL(9) =      Mean of Q
C     COBAL(10)=      Covareance of Q
C     COBAL(11)=      Air mass
C
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)

      DIMENSION Q(IMJM,KMAX),QFLUX(IMJM),
     * HTM(IMJM,KMAX),
     * U(IMJM,KMAX),V(IMJM,KMAX),VTM(IMJM,KMAX),
     * PS(IMJM),
     * IWA(LEN),COBAL(20)
C
C            1    2    3    4    5
C
C  IWA(*)=   P    LEN  NINN NWE NSN [NINN],[NWE],[NWE],[NSN],[NSN],
C            [NINN]
C
C-----------------------------------------------------------

       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,UETA(IMJM,LM),VETA(IMJM,LM),
     & TETA(IMJM,LM),QETA(IMJM,LM)
C-----------------------------------------------------------
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/ETAREA/CAREA(IMJM)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C-----------------------------------------------------------
      DATA G/9.81/
      DATA PI/3.14159265/
      DATA ERAD /6370000./
C
C     S T A R T
C
      IF(IWA(1).EQ.0)THEN
      IPJ=I0+J0
      NX1=NX+1
      NY1=NY+1
```

```fortran
      IF(MOD(IPJ,2).NE.0.OR.MOD(NX1,2).NE.0.OR.MOD(NY1,2).NE.0)THEN
      WRITE(66,*)'MISTAKE IN BALARE PARAMETERS'
      WRITE(66,*)'I0=',I0,' I0+J0=',IPJ
      WRITE(66,*)'NX=',NX,' NX+1 =',NX1
      WRITE(66,*)'J0=',J0,' NY+1 =',NY1
      WRITE(66,*)'NY=',NY
      STOP
      ENDIF

      NX2=NX/2
      NY2=NY/2

      NINN=NX*NX2+NX2+1
      NWE=NY2
      NSN=NX2
      IWA(1)=1
      IWA(2)=LEN
      IWA(3)=NINN
      IWA(4)=NWE
      IWA(5)=NSN

      LEN1=5+NINN+NWE+NWE+NSN+NSN+NINN

      IF(LEN1.GT.LEN)THEN
      WRITE(66,*)'ERROR IN BALARE: NO MEMORY'
      WRITE(66,*)'LEN=',LEN,' LEN1=',LEN1
      STOP
      ENDIF

      IBASE=5
      INNER=IBASE
      IWB=IBASE+NINN
      IEB=IWB+NWE
      ISB=IEB+NWE
      INB=ISB+NSN
      IAJ=INB+NSN       ! Latitude numbers of the points

      NXX=I0+NX-1
      NYY=J0+NY-1
      K=0
      DO J=J0,NYY
      DO I=I0,NXX
      IPJ=I+J
      IF(MOD(IPJ,2).EQ.0)THEN
      K=K+1
      IWA(INNER+K)=INDEXH(I,J,IM)
      IWA(IAJ+K)=J
      ENDIF
      ENDDO
      ENDDO
C ==========================
C                                        !
C            W and E boundaries          !NYY=J0+NY-1
      K=0                                !    x---o---x---o---x
      DO J=J0,NYY                        !    |   |   |   |   |
      IPJ=I0+J                           !    o---x---o---x---o
      IF(MOD(IPJ,2).EQ.1)THEN            !    |   |   |   |   |
      K=K+1                              !    x---o---x---o---x
      IWA(IWB+K)=INDEXV(I0, J,IM)        !    |   |   |   |   |
      IWA(IEB+K)=INDEXV(NXX,J,IM)        !    o---x---o---x---o
      ENDIF                              !    |   |   |   |   |
      ENDDO                              ! J0 x---o---x---o---x
C            S and N boundaries          !   I0               NXX=I0+NX-1
      K=0
      DO I=I0,NXX
      IPJ=J0+I
      IF(MOD(IPJ,2).EQ.1)THEN
      K=K+1
      IWA(ISB+K)=INDEXV(I,J0, IM)
```



```fortran
      IWA(INB+K)=INDEXV(I,NYY,IM)
      ENDIF
      ENDDO
C
C     DEBUGING
C
C
C     DEBUGING
C
C/      WRITE(66,*)' DEBUGING ***************'
C/      WRITE(66,*)'IBASE=',IBASE
C/      WRITE(66,*)'INNER=',INNER
C/      WRITE(66,*)'IW  =',IW
C/      WRITE(66,*)'IE  =',IE
C/      WRITE(66,*)'IS  =',IS
C/      WRITE(66,*)'IN  =',IN
C/      WRITE(66,*)'1)',IWA(INNER+1),0,IWA(INNER+2),0,IWA(INNER+3)
C/      WRITE(66,*)'2)',0,IWA(INNER+4),0,IWA(INNER+5),0
C/      WRITE(66,*)'3)',IWA(INNER+6),0,IWA(INNER+7),0,IWA(INNER+8)
C/      WRITE(66,*)'4)',0,IWA(INNER+9),0,IWA(INNER+10),0
C/      WRITE(66,*)'5)',IWA(INNER+11),0,IWA(INNER+12),0,IWA(INNER+13)
100   FORMAT(10X,5(I5,1X))
C/      WRITE(66,*)' S '
C/      WRITE(66,*)(IWA(ISB+J),J=1,2)
C/      DO J=1,2
C/      WRITE(66,*)J,')',IWA(IWB+J),IWA(IEB+J)
C/      ENDDO
C/      WRITE(66,*)' N '
C/      WRITE(66,*)(IWA(INB+J),J=1,2)
C/
C/      WRITE(66,*)' IWA ARRAY'
C/      DO J=1,30
C/      WRITE(66,*)J,IWA(J)
C/      ENDDO
C/
      ENDIF
C     -----
C     C O N T I N U E
C     -----
      NINN=IWA(3)
      NWE=IWA(4)
      NSN=IWA(5)

      IBASE=5
      INNER=IBASE
      IWB=IBASE+NINN
      IEB=IWB+NWE
      ISB=IEB+NWE
      INB=ISB+NSN

C/      WRITE(66,*)' DEBUG   ***************'
C/      WRITE(66,*)'INNER=',INNER
C/      WRITE(66,*)'IWB  =',IWB
C/      WRITE(66,*)'IEB  =',IEB
C/      WRITE(66,*)'ISB  =',ISB
C/      WRITE(66,*)'INB  =',INB
C/      WRITE(66,*)'NINN =',NINN
C/      WRITE(66,*)'NWE =',NWE
C/      WRITE(66,*)'NSN =',NSN
C
      CON=PI/180.
C
C     Latitude and longitude steps are converted in radians
C
      DX1=2.*CON*DEX
      DY1=2.*CON*DEY
C
C     WEST AND EAST
C
```

```
      DQW=0.
      DQE=0.
      DO L=1,KMAX

      DO J=1,NWE

      K=IWA(IWB+J)
      UW=U(K,L)*VTM(K,L)
      DP= DETA(L)*PS(K)*RES(K)
      AIR=DP/G

      K1=K+IM          ! WARNING: MISTAKE IN FLUXMS
      K2=K-IM+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)*HTM(K1,L)+Q(K2,L)*HTM(K2,L))*WET(IW)
      DQW=DQW+UW*Q1*DY1*AIR*ERAD

C/      WRITE(66,*)'K,K1,K2=',K,K1,K2,' L=',L,' J=',J
C/      WRITE(66,*)'H=',H,'Q1=',Q1,'UW=',UW
C/      WRITE(66,*)' '

      K=IWA(IEB+J)
      UE=-U(K,L)*VTM(K,L)
      DP= DETA(L)*PS(K)*RES(K)
      AIR=DP/G

      K1=K+IM
      K2=K-IM+1
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)*HTM(K1,L)+Q(K2,L)*HTM(K2,L))*WET(IW)
      DQE=DQE+UE*Q1*DY1*AIR*ERAD

      ENDDO
      ENDDO
C
C     SOUTH AND NORTH
C
      DQS=0.
      DQN=0.
      JSOUTH=J0
      JNORTH=J0+NY-1

      DO L=1,KMAX

      DO J=1,NSN

      K=IWA(ISB+J)
      VS=V(K,L)*VTM(K,L)
      DP= DETA(L)*PS(K)*RES(K)
      AIR=DP/G

      K1=K+1     !  WARRNING
      K2=K
      H=HTM(K1,L)+HTM(K2,L)
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)*HTM(K1,L)+Q(K2,L)*HTM(K2,L))*WET(IW)
      DQS=DQS+VS*Q1*DX1*AIR*COSETA(JSOUTH)*ERAD

      K=IWA(INB+J)
      VN=-V(K,L)*VTM(K,L)
      DP= DETA(L)*PS(K)*RES(K)
      AIR=DP/G

      K1=K+1     !  WARRNING
      K2=K
      H=HTM(K1,L)+HTM(K2,L)
```



```
      IW=INT(H+0.5)+1
      Q1=(Q(K1,L)*HTM(K1,L)+Q(K2,L)*HTM(K2,L))*WET(IW)
      DQN=DQN+VN*Q1*DX1*AIR*COSETA(JNORTH)*ERAD

      ENDDO
      ENDDO
      DELQ=DT*(DQW+DQE+DQS+DQN)

      COBAL(3)=DELQ
      COBAL(4)=DT*DQS
      COBAL(5)=DT*DQE
      COBAL(6)=DT*DQN
      COBAL(7)=DT*DQW

      TOTM=0.

      AIRMA=0.
      DO L=1,LM
      DO J=1,NINN
      K=IWA(INNER+J)
      DP= DETA(L)*PS(K)*RES(K)
      AIR=DP/G
      H=HTM(K,L)
      TOTM=TOTM+Q(K,L)*AIR*CAREA(K)*H
      AIRMA=AIRMA+AIR*CAREA(K)*H
      ENDDO
      ENDDO

      COBAL(1)=TOTM
      QMEAN=TOTM/AIRMA       !   ppm
      COBAL(9)=QMEAN
      COBAL(11)=AIRMA

      COVQ=0.
      POIN=0.
      DO L=1,LM
      DO J=1,NINN
      K=IWA(INNER+J)
      H=HTM(K,L)
      POIN=POIN+H
      COVQ=COVQ+H*(Q(K,L)-QMEAN)**2
      ENDDO
      ENDDO

      COVQ=COVQ/(POIN-1.)
      COBAL(10)=COVQ

      FBOT=0.
      SA=0.
      DO J=1,NINN
      K=IWA(INNER+J)
      FBOT=FBOT+QFLUX(K)*CAREA(K)
      SA=SA+CAREA(K)
      ENDDO

      COBAL(2)=DT*FBOT    ! 1.E6 kg/step
      COBAL(8)=SA



      RETURN
      END
C& BNDAR
      SUBROUTINE BNDAR(NU,I0,NX,J0,NY,MK)
C
C     Subroutine calculates and writes
C     bounday coordinates
C
CX       C O M M O N /MAPOT/
```

```
CX      & LSL,IXM,IYM
CX      &,ISHDE(49)
CX      &,R,PT,TSPH
CX      &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD
CX      &,CMLD,DP30,X1P,Y1P
CX      &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
CX      &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
CX      &,ERLAM0,CPHI0,SPHI0
CX      &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C23456
        RADDEG=3.14159265/180.
        DEG=180./3.14159265

      WRITE(NU,*)NX*NY
      IEND=I0+NX-1
      JEND=J0+NY-1

      J=J0
      DO I=I0,IEND
      CALL GCOORD(I,J,GLON,GLAT)
      WRITE(NU,*)I,J,GLAT,GLON,MK
      ENDDO

      I=IEND
      DO J=J0,JEND
      CALL GCOORD(I,J,GLON,GLAT)
      WRITE(NU,*)I,J,GLAT,GLON,MK
      ENDDO

      J=JEND
      DO I=IEND,I0,-1
      CALL GCOORD(I,J,GLON,GLAT)
      WRITE(NU,*)I,J,GLAT,GLON,MK
      ENDDO

      I=I0
      DO J=JEND,J0,-1
      CALL GCOORD(I,J,GLON,GLAT)
      WRITE(NU,*)I,J,GLAT,GLON,MK
      ENDDO

      WRITE(NU,'(''end'')')
      RETURN
      END
C& LOCADR
      SUBROUTINE LOCADR(NU,IWA,NX,NY)
      DIMENSION IWA(1)
      DIMENSION IA(1000)
      IF(IWA(1).EQ.0)THEN
      WRITE(NU,*)'IWA ia empty'
      RETURN
      ENDIF
      K=0
      NINN=IWA(3)
      INNER=5
      DO J=1,NINN
        K=K+1
        IA(K)=IWA(INNER+J)
        IF(J.NE.NINN)THEN
        K=K+1
        IA(K)=0
        ENDIF
      ENDDO
      WRITE(NU,*)' '
      DO J=NY,1,-1
      IAD=(J-1)*NX
      WRITE(NU,100)J,(IA(IAD+I),I=1,NX)
      ENDDO
      WRITE(NU,*)' '
```



```
      RETURN
100   FORMAT(2X,I3,1X,30(I5,1X))
      END
C& LATFLX
      SUBROUTINE LATFLX(Q,HTM,U,V,VTM,PS,IWA,LEN,FLXL)
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      PARAMETER (IMAX=2*IM-1  , JMAX=JM)
      PARAMETER (KMAX=LM)
      DIMENSION Q(IMJM,KMAX),HTM(IMJM,KMAX),
     * U(IMJM,KMAX),V(IMJM,KMAX),VTM(IMJM,KMAX),
     * PS(IMJM),IWA(LEN)
C
C            1    2    3    4   5
C
C  IWA(*)=   P   LEN  NINN NWE NSN [NINN],[NWE],[NWE],[NSN],[NSN],
C            [NINN]
C
      DIMENSION NORM(2,4), KH(5), KV(5), DS(4), COSFI(4)
      REAL NORM
C-----------------------------------------------------------

       C O M M O N /MAPOT/
     & LSL,IXM,IYM
     &,ISHDE(49)
     &,R,PT,TSPH
     &,WBD,SBD,TLM0D,TPH0D,DLMD,DPHD,CMLD,DP30,X1P,Y1P
     &,DISLP,Z0SLP,SPL(LSM),ALSL(LSM)
     &,DETA(LM),AETA(LM),ETA(LP1),TSHDE(49)
     &,ERLAM0,CPHI0,SPHI0
     &,KUNIT,LUNIT,MUNIT,IUNIT,NUNIT
C-----------------------------------------------------------
       C O M M O N /VRBLS/
     & PD(IMJM),RES(IMJM),FIS(IMJM)
C
     &,UETA(IMJM,LM),VETA(IMJM,LM),
     & TETA(IMJM,LM),QETA(IMJM,LM)
C-----------------------------------------------------------
      PARAMETER (NPOINT=25)
      COMMON/ETAGRD/
     *      XB,DEX,YB,DEY,
     *      COSETA(JM),XETA(IMAX),YETA(JM),
     *      ITMP(NPOINT), JTMP(NPOINT), KTMP(NPOINT)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)


Cx      DATA NORM/ 0.70710678,-0.70710678,
Cx     *          -0.70710678,-0.70710678,
Cx     *          -0.70710678, 0.70710678,
Cx     *           0.70710678, 0.70710678/
      DATA NORM/ 1.0,-1.0,
     *          -1.0,-1.0,
     *          -1.0, 1.0,
     *           1.0, 1.0/
      DATA ERAD /6370000./
      DATA G /9.81/
      DATA CON/-999.0/
      DATA PI/3.14159265/
C
C                KH3   KV3   KH2
C                       / \
C                      /   \
C                     /     \
C                    /       \
C                KV4    K    KV2
C                    \       /
C                     \     /
C                      \   /
```

```
C                    \ /
C               KH4   KV1   KH1
C                     KV5

      SF=0.
      NINN=IWA(3)
      IBASE=5+NINN+2*(IWA(4)+IWA(5))
      DX=PI*DEX/180.
      DY=PI*DEY/180.

      DO I=1,NINN

      K=IWA(5+I)
      KH0=K
      KH(4)=K-IM
      KH(1)=KH(4)+1
      KH(2)=K+IM
      KH(3)=KH(2)-1
      KH(5)=K

      KV(2)=K
      KV(4)=KV(2)-1
      KV(1)=KH(4)
      KV(3)=KH(3)
      KV(5)=KV(1)

      J0=IWA(IBASE+I)
      COSFI(1)=COSETA(J0-1)
      COSFI(2)=COSETA(J0)
      COSFI(3)=COSETA(J0)
      COSFI(4)=COSETA(J0-1)

      IH0 =KH(5)

      DS(1)=ERAD*SQRT(DX**2+(COSFI(1)*DY)**2)
      DS(2)=ERAD*SQRT(DX**2+(COSFI(2)*DY)**2)
      DS(3)=ERAD*SQRT(DX**2+(COSFI(3)*DY)**2)
      DS(4)=ERAD*SQRT(DX**2+(COSFI(4)*DY)**2)

      DO L=1,LM

      DPQ0=DETA(L)*PS(IH0)*RES(IH0)*Q(IH0,L)*HTM(IH0,L)/G

      DO IB=1,4
      IH  =KH(IB)
      IV  =KV(IB)
      IVP1=KV(IB+1)
      H=VTM(IV,L)+VTM(IVP1,L)
      IW=INT(H+0.5)+1
      UI=(U(IV,L)+U(IVP1,L))*WET(IW)
      VI=(V(IV,L)+V(IVP1,L))*WET(IW)
      VN=NORM(1,IB)*UI+NORM(2,IB)*VI

      H=HTM(IH0,L)+HTM(IH,L)
      IW=INT(H+0.5)+1
      DPQI=DETA(L)*PS(IH)*RES(IH)*Q(IH,L)*HTM(IH,L)/G
      AMI=0.5*(DPQ0+DPQI)*WET(IW)
      FLXI=VN*AMI*DS(IB)
      SF=SF+FLXI
      ENDDO  ! ......................  IB loop from 1 to 4

      ENDDO  ! ......................  Level loop
      ENDDO  ! ......................  Inner point loop

      FLXL=SF
      RETURN
      END
C& ENFILT
      SUBROUTINE ENFILT(Q,QE)
```



```
      PARAMETER (IM=111,JM=221,LM=38,LSM=19)
      PARAMETER (IMJM = IM*JM - JM/2)
      PARAMETER (LP1=LM+1)
      DIMENSION Q(IMJM,LM), QE(IMJM,LM)
      C O M M O N /MASKS/
     & HBM2(IMJM),VBM2(IMJM),VBM3(IMJM),SM(IMJM),SICE (IMJM)
     &,HTM(IMJM,LM),VTM(IMJM,LM)
      COMMON/INNERP/IPN,INP(IMJM),IINP(IMJM),JINP(IMJM)
      COMMON/TAB25A/IOA(25),WET(26),IOAC(16)
C
      AL1=0.75
      AL2=0.25
      DO J=1,IPN
      K=INP(J)
      K2=K+IM
      K3=K2-1
      K4=K-IM
      K1=K4+1

      DO L=1,LM

      H0=HTM(K,L)
      H=HTM(K1,L)+HTM(K2,L)+HTM(K3,L)+HTM(K4,L)
      IW=INT(H+0.5)+1
      Q1=AL1*Q(K,L)
      Q2=HTM(K1,L)*Q(K1,L)+HTM(K2,L)*Q(K2,L)+
     *   HTM(K3,L)*Q(K3,L)+HTM(K4,L)*Q(K4,L)
      Q2=AL2*Q2*WET(IW)

      QE(K,L)=H0*(Q1+Q2)
      ENDDO          !    Level loop

      ENDDO          !    Inner point  Loop
      RETURN
      END
C& TRDATE
      SUBROUTINE TRDATE(DATE1,DATE2)
      CHARACTER*7 DATE1
      CHARACTER*6 DATE2
      CHARACTER*3 MON1(12)
      CHARACTER*2 MON2(12)
      DATA MON1/'JAN','FEB','MAR','APR','MAI','JUN',
     *          'JUL','AUG','SEP','OCT','NOV','DEC'/
      DATA MON2/'01','02','03','04','05','06',
     *          '07','08','09','10','11','12'/

      DO I=1,12
      M=I
      IF(DATE1(3:5).EQ.MON1(I))GOTO10
      ENDDO
      WRITE(66,*)'ERROR IN TRDATE: MISTAKE WITH DATE'
      WRITE(66,*)'DATE=',DATE1
      STOP 'MISTAKE WITH DATE'
10    CONTINUE
      DATE2=DATE1(1:2)//MON2(M)//DATE1(6:7)
      RETURN
      END
C& TR2DATE
      SUBROUTINE TR2DATE(IDAT,DATE,DATE2)
      DIMENSION IDAT(3)
      CHARACTER*6 DATE2
      CHARACTER*7 DATE
      CHARACTER*3 MON1(12)
      CHARACTER*2 MON2(12), DAY(31), YEAR(20)
      DATA MON1/'JAN','FEB','MAR','APR','MAI','JUN',
     *          'JUL','AUG','SEP','OCT','NOV','DEC'/
      DATA MON2/'01','02','03','04','05','06',
     *          '07','08','09','10','11','12'/
      DATA DAY/'01','02','03','04','05','06',
```

```
     *         '07','08','09','10','11','12',
     *         '13','14','15','16','17','18',
     *         '19','20','21','22','23','24',
     *         '25','26','27','28','29','30',
     *         '31'/
      DATA YEAR/'80','81','82','83','84',
     *          '85','86','87','88','89',
     *          '90','91','92','93','94',
     *          '95','96','97','98','99'/

      ID=IDAT(2)
      IM=IDAT(1)
      IY=IDAT(3)-1979
      IF(IY.LT.1.OR.IY.GT.20)THEN
      WRITE(66,*)'ERROR IN INITIAL DATE:'
      WRITE(66,*)'IDAT=',IDAT
      WRITE(66,*)'Year.GE.1980.and.Year.LE.1999'
      STOP
      ENDIF

      DATE=DAY(ID)//MON1(IM)//YEAR(IY)
      DATE2=DAY(ID)//MON2(IM)//YEAR(IY)
      RETURN
      END
C& CALEND
      SUBROUTINE CALEND(DMYH,NDAY)
C
C     The day number of DATE in year
C
C     DMYH=20089500
C          ddmmyyhh
C     NDAY=Julian day
C
      INTEGER DMYH,DDMMYY,HH
      DIMENSION CL(12,2),CC(12,2)
      INTEGER DD,YY,CL,CC
      DATA CL /31,28,31,30,31,30,
     *         31,31,30,31,30,31,
     *         31,29,31,30,31,30,
     *         31,31,30,31,30,31/
      DATA CC / 1, 32, 60, 91,121,152,
     *        182,213,244,274,305,335,
     *          1, 32, 61, 92,122,153,
     *        183,214,245,275,306,336/
      DDMMYY=DMYH/100
      HH=MOD(DMYH,100)
      YY=1900+MOD(DDMMYY,100)
      DD=(DDMMYY/10000)
      MM=(DDMMYY/100-100*DD)
      IF(MOD(YY,4).EQ.0)THEN
      K=2
      ELSE
      K=1
      ENDIF
      NDAY=CC(MM,K)+DD-1
      RETURN
      END
```