Appendix E. FORTRAN Language Program To Read an Archive Record

```
program testRead(arcfile,outfile)

      ! The program takes two arguments, specifying the
input
      ! filename and the output filename (including paths.)
      ! The output file is written in the current working
directory
      ! unless a path is included as part of the filename.

      integer*2 j
      integer*4 rec,eof,hr,min,sec
      character*80 arcfile,outfile

      ! Common block /rec/ variables:
      integer*4 cSum,ihd(6,4)
      integer*2
cSumFlag,major,status(10),qual(16),minor(16),mdf(40,16)
      real*4
analog(17),head(27,4),ssLoc(2,16),mep0(9,16),mep90(9,16),
     +
mepOmni(4,16),ted0(8,16),ted30(8,16),ted0s(8,4),ted30s(8,4),
     +    tedback(2,4),tedfx(7,16)

      common /rec/cSum,ihd,cSumFlag,major,status,qual,minor,
     +
mdf,analog,head,ssLoc,mep0,mep90,mepOmni,ted0,ted30,
     +    ted0s,ted30s,tedback,tedfx

      eof = 0
      open(unit=7,access='direct',recl=2544,file =
arcfile,status='old')
      open(unit=11,access='sequential',file = outfile,
     +     status='unknown',form='formatted')

      do rec = 1,3
          call readArcSub(7,rec,eof)
          if(eof.eq.1) goto 100
          do j=1,4
              hr = ihd(4,j)/3600000
              min = (ihd(4,j) - hr*3600000)/60000
              sec = (ihd(4,j) - hr*3600000 - min*60000)/1000
              write(11,200),ihd(2,j),ihd(3,j),hr,min,sec,
     +                  head(2,j),head(3,j),ihd(6,j)
          enddo
          write(11,300)"mep0:"
```

```
          do j=1,16

write(11,220)j,mep0(1,j),mep0(2,j),mep0(3,j),mep0(4,j),
     +
mep0(5,j),mep0(6,j),mep0(7,j),mep0(8,j),mep0(9,j)
          enddo
          write(11,300)"mep90:"
          do j=1,16

write(11,220)j,mep90(1,j),mep90(2,j),mep90(3,j),
     +
mep90(4,j),mep90(5,j),mep90(6,j),mep90(7,j),mep90(8,j),
     +          mep90(9,j)
          enddo
          write(11,300)"mepOmni:"
          do j=1,16

write(11,220)j,mepOmni(1,j),mepOmni(2,j),mepOmni(3,j),
     +       mepOmni(4,j)
          enddo
          write(11,300)"ted0:"
          do j=1,16

write(11,220)j,ted0(1,j),ted0(2,j),ted0(3,j),ted0(4,j),
     +          ted0(5,j),ted0(6,j),ted0(7,j),ted0(8,j)
          enddo
          write(11,300)"ted30:"
          do j=1,16

write(11,220)j,ted30(1,j),ted30(2,j),ted30(3,j),
     +          ted30(4,j),ted30(5,j),ted30(6,j),ted30(7,j),
     +          ted30(8,j)
          enddo
          write(11,300)"ted0s:"
          do j=1,4

write(11,220)j,ted0s(1,j),ted0s(2,j),ted0s(3,j),
     +
ted0s(4,j),ted0s(5,j),ted0s(6,j),ted0s(7,j),ted0s(8,j)
          enddo
          write(11,300)"ted30s:"
          do j=1,4

write(11,220)j,ted30s(1,j),ted30s(2,j),ted30s(3,j),
     +
ted30s(4,j),ted30s(5,j),ted30s(6,j),ted30s(7,j),
     +          ted30s(8,j)
```

```
          enddo
          write(11,300)"tedback:"
          do j=1,4
              write(11,220)j,tedback(1,j),tedback(2,j)
          enddo
          write(11,300)"tedfx:"
          do j=1,16

write(11,220)j,tedfx(1,j),tedfx(2,j),tedfx(3,j),
     +           tedfx(4,j),tedfx(5,j),tedfx(6,j),tedfx(7,j)
          enddo

          write(11,300)"-----------------------------------
----------"

      enddo

  200 format(i4,1x,i3,3(1x,i2),1x,f8.3,1x,f8.3,1x,i7)
  220 format(i4,9(1x,f8.2))
  300 format(A)
  100 close(7)
      close(11)
      end


subroutine readArcSub (unit,rnum,eof)

      ! ------------------------------------------------
-------
      !   readArcSub.f returns a NOAA POES SEM2 data record
from an
      !               already opened file in packed binary
format.
      !
      !   Calling from C:  It is recommended that C users
use the
      !                    C routine: unpackSem2.c instead.
      !
      !   Compilation: f77 -c readArcSub.f
      !
      !
      !   Programmer:  Sue Greer
      !
      ! ------------------------------------------------
-------
      byte arcBytes(2544)
      integer*4 unit,rnum,eof,ios
```

```
      integer*4 arc(2544),i,j,jj,k,m,mf,mo,mp0,mp90,qf,tf
      integer*4 b1,b2,b3,b4
      integer*2 onebyte
      real*4 cf ! Conversion factor (.0001)
      real*4 cnvrt(256)

      ! Common block /rec/ variables:
      integer*4 cSum,ihd(6,4)
      integer*2
cSumFlag,major,status(10),qual(16),minor(16),mdf(40,16)
      real*4
analog(17),head(27,4),ssLoc(2,16),mep0(9,16),mep90(9,16),
     +
mepOmni(4,16),ted0(8,16),ted30(8,16),ted0s(8,4),ted30s(8,4),
     +    tedback(2,4),tedfx(7,16)

      common /rec/cSum,ihd,cSumFlag,major,status,qual,minor,
     +
mdf,analog,head,ssLoc,mep0,mep90,mepOmni,ted0,ted30,
     +    ted0s,ted30s,tedback,tedfx

      cf = 0.0001
      b1=256*256*256
      b2=256*256
      b3=256
      b4=1

      data cnvrt
/0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0,11.0,

+12.0,13.0,14.0,15.0,16.0,17.0,18.0,19.0,20.0,21.0,22.0,23.0
,24.0,

+25.0,26.0,27.0,28.0,29.0,30.0,31.0,32.0,34.5,36.5,38.5,40.5
,42.5,

+44.5,46.5,48.5,50.5,53.0,56.0,59.0,62.0,65.5,69.5,73.5,77.5
,81.5,

+85.5,89.5,93.5,97.5,101.5,106.5,112.5,118.5,124.5,131.5,139
.5,

+147.5,155.5,163.5,171.5,179.5,187.5,195.5,203.5,213.5,225.5
,

+237.5,249.5,263.5,279.5,295.5,311.5,327.5,343.5,359.5,375.5
,
```

```
+391.5,407.5,427.5,451.5,475.5,499.5,527.5,559.5,591.5,623.5
,

+655.5,687.5,719.5,751.5,783.5,815.5,855.5,903.5,951.5,999.5
,

+1055.5,1119.5,1183.5,1247.5,1311.5,1375.5,1439.5,1503.5,156
7.5,

+1631.5,1711.5,1807.5,1903.5,1999.5,2111.5,2239.5,2367.5,249
5.5,

+2623.5,2751.5,2879.5,3007.5,3135.5,3263.5,3423.5,3615.5,380
7.5,

+3999.5,4223.5,4479.5,4735.5,4991.5,5247.5,5503.5,5759.5,601
5.5,

+6271.5,6527.5,6847.5,7231.5,7615.5,7999.5,8447.5,8959.5,947
1.5,

+9983.5,10495.5,11007.5,11519.5,12031.5,12543.5,13055.5,1369
5.5,

+14463.5,15231.5,15999.5,16895.5,17919.5,18943.5,19967.5,209
91.5,

+22015.5,23039.5,24063.5,25087.5,26111.5,27391.5,28927.5,304
63.5,

+31999.5,33791.5,35839.5,37887.5,39935.5,41983.5,44031.5,460
79.5,

+48127.5,50175.5,52223.5,54783.5,57855.5,60927.5,63999.5,675
83.5,

+71679.5,75775.5,79871.5,83967.5,88063.5,92159.5,96255.5,100
351.5,

+104447.5,109567.5,115711.5,121855.5,127999.5,135167.5,14335
9.5,

+151551.5,159743.5,167935.5,176127.5,184319.5,192511.5,20070
3.5,
```

```
+208895.5,219135.5,231423.5,243711.5,255999.5,270335.5,28671
9.5,

+303103.5,319487.5,335871.5,352255.5,368639.5,385023.5,40140
7.5,

+417791.5,438271.5,462847.5,487423.5,511999.5,540671.5,57343
9.5,

+606207.5,638975.5,671743.5,704511.5,737279.5,770047.5,80281
5.5,

+835583.5,876543.5,925695.5,974847.5,1023999.5,1081343.5,

+1146879.5,1212415.5,1277951.5,1343487.5,1409023.5,1474559.5
,

+1540095.5,1605631.5,1671167.5,1753087.5,1851391.5,1949695.5
,
     +1998848.0/

      eof = 0
      if (rnum.lt.1) goto 100

   10 read(unit,rec=rnum,iostat=ios) arcBytes
      if(ios.ne.0) then
          eof = 1
          goto 100
      endif
      do i=1,2544
          ! Byte type goes from -128 to +127
          ! Convert to 8-bit integer
          onebyte = arcBytes(i)
          arc(i) = onebyte
          if(onebyte.lt.0) arc(i) = onebyte+256
      enddo

* ------------------------------------------------------------
--------
*     ! Parse the data in arc()
* ------------------------------------------------------------
--------

      ! Checksum flag
      cSumFlag = arc(1)*b1 + arc(2)*b2 + arc(3)*b3 + arc(4)
      ! Checksum
```

```
      cSum = arc(5)*b1 + arc(6)*b2 + arc(7)*b3 + arc(8)
      ! Major frame number
      major = arc(9)*b3 + arc(10)*b4
      !print *,'cSumFlag, cSum, MF ',cSumFlag,cSum,major

      ! Status: instrument on/off, ifc, etc.
      do i = 1,10
          status(i) = arc(i+10)
          !print *,i,status(i)
      enddo

      ! Analog data, etc. 4-byte integers converted to reals

      do i = 1,68,4
          k = (i+1)/4 + 1
          analog(k) =
(arc(i+20)*b1+arc(i+21)*b2+arc(i+22)*b3+
     +         arc(i+23))*cf
          !print *,k,analog(k)
      enddo

      ! Sub-satellite location, 4-byte integers converted to
reals
      k = 1
      ssLoc(1,1) = (arc(k+88)*b1+arc(k+89)*b2+arc(k+90)*b3+
     +         arc(k+91))*cf
      ssLoc(1,2) = (arc(k+92)*b1+arc(k+93)*b2+arc(k+94)*b3+
     +         arc(k+95))*cf
      ssLoc(1,3) = (arc(k+96)*b1+arc(k+97)*b2+arc(k+98)*b3+
     +         arc(k+99))*cf
      ssLoc(1,4) =
(arc(k+100)*b1+arc(k+101)*b2+arc(k+102)*b3+
     +         arc(k+103))*cf
      ssLoc(2,1)
=(arc(k+104)*b1+arc(k+105)*b2+arc(k+106)*b3+
     +         arc(k+107))*cf
      ssLoc(2,2)
=(arc(k+108)*b1+arc(k+109)*b2+arc(k+110)*b3+
     +         arc(k+111))*cf
      ssLoc(2,3)
=(arc(k+112)*b1+arc(k+113)*b2+arc(k+114)*b3+
     +         arc(k+115))*cf
      ssLoc(2,4)
=(arc(k+116)*b1+arc(k+117)*b2+arc(k+118)*b3+
     +         arc(k+119))*cf
      ssLoc(1,5) =
(arc(k+120)*b1+arc(k+121)*b2+arc(k+122)*b3+
```

```
     +         arc(k+123))*cf
      ssLoc(1,6) =
(arc(k+124)*b1+arc(k+125)*b2+arc(k+126)*b3+
     +         arc(k+127))*cf
      ssLoc(1,7) =
(arc(k+128)*b1+arc(k+129)*b2+arc(k+130)*b3+
     +         arc(k+131))*cf
      ssLoc(1,8) =
(arc(k+132)*b1+arc(k+133)*b2+arc(k+134)*b3+
     +         arc(k+135))*cf
      ssLoc(2,5)
=(arc(k+136)*b1+arc(k+137)*b2+arc(k+138)*b3+
     +         arc(k+139))*cf
      ssLoc(2,6)
=(arc(k+140)*b1+arc(k+141)*b2+arc(k+142)*b3+
     +         arc(k+143))*cf
      ssLoc(2,7)
=(arc(k+144)*b1+arc(k+145)*b2+arc(k+146)*b3+
     +         arc(k+147))*cf
      ssLoc(2,8)
=(arc(k+148)*b1+arc(k+149)*b2+arc(k+150)*b3+
     +         arc(k+151))*cf
      ssLoc(1,9) =
(arc(k+152)*b1+arc(k+153)*b2+arc(k+154)*b3+
     +         arc(k+155))*cf
      ssLoc(1,10) =
(arc(k+156)*b1+arc(k+157)*b2+arc(k+158)*b3+
     +         arc(k+159))*cf
      ssLoc(1,11) =
(arc(k+160)*b1+arc(k+161)*b2+arc(k+162)*b3+
     +         arc(k+163))*cf
      ssLoc(1,12) =
(arc(k+164)*b1+arc(k+165)*b2+arc(k+166)*b3+
     +         arc(k+167))*cf
      ssLoc(2,9)
=(arc(k+168)*b1+arc(k+169)*b2+arc(k+170)*b3+
     +         arc(k+171))*cf
      ssLoc(2,10)
=(arc(k+172)*b1+arc(k+173)*b2+arc(k+174)*b3+
     +         arc(k+175))*cf
      ssLoc(2,11)
=(arc(k+176)*b1+arc(k+177)*b2+arc(k+178)*b3+
     +         arc(k+179))*cf
      ssLoc(2,12)
=(arc(k+180)*b1+arc(k+181)*b2+arc(k+182)*b3+
     +         arc(k+183))*cf
```

```
      ssLoc(1,13) =
(arc(k+184)*b1+arc(k+185)*b2+arc(k+186)*b3+
     +         arc(k+187))*cf
      ssLoc(1,14) =
(arc(k+188)*b1+arc(k+189)*b2+arc(k+190)*b3+
     +         arc(k+191))*cf
      ssLoc(1,15) =
(arc(k+192)*b1+arc(k+193)*b2+arc(k+194)*b3+
     +         arc(k+195))*cf
      ssLoc(1,16) =
(arc(k+196)*b1+arc(k+197)*b2+arc(k+198)*b3+
     +         arc(k+199))*cf
      ssLoc(2,13)
=(arc(k+200)*b1+arc(k+201)*b2+arc(k+202)*b3+
     +         arc(k+203))*cf
      ssLoc(2,14)
=(arc(k+204)*b1+arc(k+205)*b2+arc(k+206)*b3+
     +         arc(k+207))*cf
      ssLoc(2,15)
=(arc(k+208)*b1+arc(k+209)*b2+arc(k+210)*b3+
     +         arc(k+211))*cf
      ssLoc(2,16)
=(arc(k+212)*b1+arc(k+213)*b2+arc(k+214)*b3+
     +         arc(k+215))*cf


* -----------------------------------------------------
*      ! Header information
* -----------------------------------------------------
      ! Subsatellite latitude and longitude
      head(2,1) = ssLoc(1,1)
      head(2,2) = ssLoc(1,5)
      head(2,3) = ssLoc(1,9)
      head(2,4) = ssLoc(1,13)
      head(3,1) = ssLoc(2,1)
      head(3,2) = ssLoc(2,5)
      head(3,3) = ssLoc(2,9)
      head(3,4) = ssLoc(2,13)

      do j = 1,4
          m = 3
          i = 1640+(j-1)*96
          do k = 4,96,4
              m = m+1
              head(m,j) = (arc(i+1+k-4)*b1 + arc(i+2+k-4)*b2
+
     +            arc(i+3+k-4)*b3 + arc(i+4+k-4))*cf
```

```
          enddo
       enddo

       j = 1    ! Index for 8-second records
       ! ihd info, 4-byte integers  (1..5,1)
       do i=1,20,4
           k = (i+1)/4 +1
           ihd(k,j) =
(arc(i+216)*b1+arc(i+217)*b2+arc(i+218)*b3+
     +         arc(i+219))
       enddo
       ! Inclination (head(1,1))
       head(1,j) =  (arc(237)*b1+arc(238)*b2+arc(239)*b3+
     +         arc(240))*cf/10.
       ! Orbit number (6,1)
       ihd(6,j) = (arc(241)*b1+arc(242)*b2+arc(243)*b3+
     +         arc(244))
       ! Quality, 2-byte integer (1..4)
       qf = 1
       do i=1,7,2
           qual(qf) = arc(i+244)*b3+arc(i+245)
           qf = qf + 1
       enddo
       ! Minor frame numbers (1..4)
       mf = 1
       do i = 1,7,2
           minor(mf) = arc(i+252)*b3+arc(i+253)
           mf = mf+1
       enddo
       ! Missing data flags (1..40,1..4)
       do i = 1,4
         do n = 1,40
               mdf(n,i) = arc(n-1+40*(i-1)+261)
         enddo
       enddo

       j = 2
       ! ihd info, 4-byte integers (1..5,2)
       do i=1,20,4
           k = (i+1)/4 +1
           ihd(k,j) =
(arc(i+420)*b1+arc(i+421)*b2+arc(i+422)*b3+
     +         arc(i+423))
       enddo
       ! Inclination (head(1,2))
       head(1,j) =  (arc(441)*b1+arc(442)*b2+arc(443)*b3+
     +         arc(444))*cf/10.
```

```
       ! Orbit number  (ihd(6,2))
       ihd(6,j) = (arc(445)*b1+arc(446)*b2+arc(447)*b3+
      +         arc(448))
       ! Quality, 2-byte integer (5..8)
       do i=1,7,2
           qual(qf) = arc(i+448)*b3+arc(i+449)
           qf = qf + 1
       enddo
       ! Minor frame numbers (5..8)
       do i = 1,7,2
           minor(mf) = arc(i+456)*b3+arc(i+457)
           mf = mf+1
       enddo
       ! Missing data flags (1..40,5..8)
       do i = 5,8
           do n = 1,40
               mdf(n,i) = arc(n-1+40*(i-5)+465)
           enddo
       enddo

       j = 3
       ! ihd info, 4-byte integers (1..5,3)
       do i=1,20,4
           k = (i+1)/4 +1
           ihd(k,j) =
  (arc(i+624)*b1+arc(i+625)*b2+arc(i+626)*b3+
      +         arc(i+627))
       enddo
       ! Inclination (head(1,3))
       head(1,j) =  (arc(645)*b1+arc(646)*b2+arc(647)*b3+
      +         arc(648))*cf/10.
       ! Orbit number  (ihd(6,3))
       ihd(6,j) = (arc(649)*b1+arc(650)*b2+arc(651)*b3+
      +         arc(652))
       ! Quality, 2-byte integer (9..12)
       do i=1,7,2
           qual(qf) = arc(i+652)*b3+arc(i+653)
           qf = qf + 1
       enddo
       ! Minor frame numbers (9..12)
       do i = 1,7,2
           minor(mf) = arc(i+660)*b3+arc(i+661)
           mf = mf+1
       enddo
       ! Missing data flags (1..40,9..12)
       do i = 9,12
           do n = 1,40
```

```
              mdf(n,i) = arc(n-1+40*(i-9)+669)
            enddo
          enddo

          j = 4
          ! ihd info, 4-byte integers (1..5,4)
          do i=1,20,4
              k = (i+1)/4 +1
              ihd(k,j) =
    (arc(i+828)*b1+arc(i+829)*b2+arc(i+830)*b3+
         +         arc(i+831))
          enddo
          ! Inclination (head(1,4))
          head(1,j) =  (arc(849)*b1+arc(850)*b2+arc(851)*b3+
         +         arc(852))*cf/10.
          ! Orbit number (ihd(6,4))
          ihd(6,j) = (arc(853)*b1+arc(854)*b2+arc(855)*b3+
         +         arc(856))
          ! Quality, 2-byte integer (13..16)
          do i=1,7,2
              qual(qf) = arc(i+856)*b3+arc(i+857)
              qf = qf + 1
          enddo
          ! Minor frame numbers (13..16)
          do i = 1,7,2
              minor(mf) = arc(i+864)*b3+arc(i+865)
              mf = mf+1
          enddo
          ! Missing data flags (1..40,13..16)
          do i = 13,16
              do n = 1,40
                  mdf(n,i) = arc(n-1+40*(i-13)+873)
              enddo
          enddo

    * -------------------------------------------------------
    *      ! MEPED sensor data
    * -------------------------------------------------------
          j = 1
          mp0 = 1
          ! Meped 0-deg telescope (1..9,1..4)
          do i = 1,4
              jj = (j-1)*4+i
              do k = 1,9
                  if(mdf(k+1,jj).eq.1) then
                      mep0(k,jj) = -999.
                  else
```

```
                mep0(k,jj) = cnvrt(1+arc(mp0+1032))
            endif
            mp0 = mp0+1
        enddo
    enddo
    ! Meped 90-deg telescope (1..9,1..4)
    mp90 = 1
    do i = 1,4
        jj = (j-1)*4+i
        do k = 1,9
            if(mdf(k+10,jj).eq.1) then
                mep90(k,jj) = -999.
            else
                mep90(k,jj) = cnvrt(1+arc(mp90+1068))
            endif
            mp90 = mp90+1
        enddo
    enddo
    ! Meped omnidirectional (1..4,1..4)
    mo = 1
    do i = 1,4
        jj = (j-1)*4+i
        do k = 1,4
            mepOmni(k,jj) = cnvrt(1+arc(mo+1104))
            mo = mo+1
        enddo
        if(mdf(20,jj).eq.1) mepOmni(1,jj) = -999.
        if(mdf(21,jj).eq.1) mepOmni(2,jj) = -999.
        if(mod(jj,2).eq.1.AND.mdf(22,jj).eq.1)
mepOmni(3,jj) = -999.
        if(mod(jj,2).eq.0.AND.mdf(22,jj).eq.1)
mepOmni(4,jj) = -999.
    enddo

    j = 2
    mp0 = 1
    ! Meped 0-deg telescope (1..9,5..8)
    do i = 1,4
        jj = (j-1)*4+i
        do k = 1,9
            if(mdf(k+1,jj).eq.1) then
                mep0(k,jj) = -999.
            else
                mep0(k,jj) = cnvrt(1+arc(mp0+1184))
            endif
            mp0 = mp0+1
        enddo
```

```
      enddo
      ! Meped 90-deg telescope (1..9,5..8)
      mp90 = 1
      do i = 1,4
          jj = (j-1)*4+i
          do k = 1,9
              if(mdf(k+10,jj).eq.1) then
                  mep90(k,jj) = -999.
              else
                  mep90(k,jj) = cnvrt(1+arc(mp90+1220))
              endif
              mp90 = mp90+1
          enddo
      enddo
      ! Meped omnidirectional (1..4,5..8)
      mo = 1
      do i = 1,4
          jj = (j-1)*4+i
          do k = 1,4
              mepOmni(k,jj) = cnvrt(1+arc(mo+1256))
              mo = mo+1
          enddo
          if(mdf(20,jj).eq.1) mepOmni(1,jj) = -999.
          if(mdf(21,jj).eq.1) mepOmni(2,jj) = -999.
          if(mod(jj,2).eq.1.AND.mdf(22,jj).eq.1)
mepOmni(3,jj) = -999.
          if(mod(jj,2).eq.0.AND.mdf(22,jj).eq.1)
mepOmni(4,jj) = -999.
      enddo

      j = 3
      mp0 = 1
      ! Meped 0-deg telescope (1..9,9..12)
      do i = 1,4
          jj = (j-1)*4+i
          do k = 1,9
              if(mdf(k+1,jj).eq.1) then
                  mep0(k,jj) = -999.
              else
                  mep0(k,jj) = cnvrt(1+arc(mp0+1336))
              endif
              mp0 = mp0+1
          enddo
      enddo
      ! Meped 90-deg telescope (1..9,9..12)
      mp90 = 1
      do i = 1,4
```

```
            jj = (j-1)*4+i
            do k = 1,9
                if(mdf(k+10,jj).eq.1) then
                    mep90(k,jj) = -999.
                else
                    mep90(k,jj) = cnvrt(1+arc(mp90+1372))
                endif
                mp90 = mp90+1
            enddo
        enddo
        ! Meped omnidirectional (1..4,9..12)
        mo = 1
        do i = 1,4
            jj = (j-1)*4+i
            do k = 1,4
                mepOmni(k,jj) = cnvrt(1+arc(mo+1408))
                mo = mo+1
            enddo
            if(mdf(20,jj).eq.1) mepOmni(1,jj) = -999.
            if(mdf(21,jj).eq.1) mepOmni(2,jj) = -999.
            if(mod(jj,2).eq.1.AND.mdf(22,jj).eq.1)
mepOmni(3,jj) = -999.
            if(mod(jj,2).eq.0.AND.mdf(22,jj).eq.1)
mepOmni(4,jj) = -999.
        enddo

        j = 4
        mp0 = 1
        ! Meped 0-deg telescope (1..9,13..16)
        do i = 1,4
            jj = (j-1)*4+i
            do k = 1,9
                if(mdf(k+1,jj).eq.1) then
                    mep0(k,jj) = -999.
                else
                    mep0(k,jj) = cnvrt(1+arc(mp0+1488))
                endif
                mp0 = mp0+1
            enddo
        enddo
        ! Meped 90-deg telescope (1..9,13..16)
        mp90 = 1
        do i = 1,4
            jj = (j-1)*4+i
            do k = 1,9
                if(mdf(k+10,jj).eq.1) then
                    mep90(k,jj) = -999.
```

```
            else
                mep90(k,jj) = cnvrt(1+arc(mp90+1524))
            endif
            mp90 = mp90+1
        enddo
    enddo
    ! Meped omnidirectional (1..4,13..16)
    mo = 1
    do i = 1,4
        jj = (j-1)*4+i
        do k = 1,4
            mepOmni(k,jj) = cnvrt(1+arc(mo+1560))
            mo = mo+1
        enddo
        if(mdf(20,jj).eq.1) mepOmni(1,jj) = -999.
        if(mdf(21,jj).eq.1) mepOmni(2,jj) = -999.
        if(mod(jj,2).eq.1.AND.mdf(22,jj).eq.1)
mepOmni(3,jj) = -999.
        if(mod(jj,2).eq.0.AND.mdf(22,jj).eq.1)
mepOmni(4,jj) = -999.
    enddo

* ------------------------------------------------------
*       TED uncalibrated energy flux, max channel
*       response, and max channel
* ------------------------------------------------------

    ! TED 0-deg and 30-deg uncalibrated energy flux
    do k=1,4
        do i = 1,4
            j = (k-1)*4 + i  !(1..16)
            ted0(1,j) = cnvrt(1+arc(1121+(i-1)*4+(k-
1)*152))
            if(mdf(27,j).eq.1) ted0(1,j) = -999.
            ted0(2,j) = cnvrt(1+arc(1122+(i-1)*4+(k-
1)*152))
            if(mdf(29,j).eq.1) ted0(2,j) = -999.
            ted0(3,j) = cnvrt(1+arc(1123+(i-1)*4+(k-
1)*152))
            if(mdf(31,j).eq.1) ted0(3,j) = -999.
            ted0(4,j) = cnvrt(1+arc(1124+(i-1)*4+(k-
1)*152))
            if(mdf(33,j).eq.1) ted0(4,j) = -999.
            ted0(5,j) = 1+arc(1137+(i-1)*4+(k-1)*152)
            ted0(6,j) = 1+arc(1138+(i-1)*4+(k-1)*152)

            if(mdf(35,j).eq.1) then
```

```
                ted0(5,j) = -999.
                ted0(6,j) = -999.
            endif
            ted0(7,j) = cnvrt(1+arc(1139+(i-1)*4+(k-
1)*152))
            if(mdf(36,j).eq.1) ted0(7,j) = -999.
            ted0(8,j) = cnvrt(1+arc(1140+(i-1)*4+(k-
1)*152))
            if(mdf(37,j).eq.1) ted0(8,j) = -999.

            ted30(1,j) = cnvrt(1+arc(1153+(i-1)*4+(k-
1)*152))
            if(mdf(28,j).eq.1) ted30(1,j) = -999.
            ted30(2,j) = cnvrt(1+arc(1154+(i-1)*4+(k-
1)*152))
            if(mdf(30,j).eq.1) ted30(2,j) = -999.
            ted30(3,j) = cnvrt(1+arc(1155+(i-1)*4+(k-
1)*152))
            if(mdf(32,j).eq.1) ted30(3,j) = -999.
            ted30(4,j) = cnvrt(1+arc(1156+(i-1)*4+(k-
1)*152))
            if(mdf(34,j).eq.1) ted30(4,j) = -999.
            ted30(5,j) = 1+arc(1169+(i-1)*4+(k-1)*152)
            ted30(6,j) = 1+arc(1170+(i-1)*4+(k-1)*152)
            if(mdf(38,j).eq.1) then
                ted30(5,j) = -999.
                ted30(6,j) = -999.
            endif
            ted30(7,j) = cnvrt(1+arc(1171+(i-1)*4+(k-
1)*152))
            if(mdf(39,j).eq.1) ted30(7,j) = -999.
            ted30(8,j) = cnvrt(1+arc(1172+(i-1)*4+(k-
1)*152))
            if(mdf(40,j).eq.1) ted30(8,j) = -999.
        enddo
    enddo

* ------------------------------------------------
*     ! Ted spectra and background
* ------------------------------------------------
    do j = 1,4
        do i=1,8
            ted0s(i,j) = cnvrt(1+arc(i+(j-1)*8+2024))
            ted30s(i,j) = cnvrt(1+arc(i+(j-1)*8+2060))
        enddo
        tedback(1,j) = cnvrt(1+arc(j+2056))
        tedback(2,j) = cnvrt(1+arc(j+2092))
```

```
enddo
do j=1,16
    if(j.eq.1.OR.j.eq.5.OR.j.eq.9.OR.j.eq.13) then
        do i=1,4
            if(mdf(22+i,j).eq.1) then
                do k=1,4
                    ted0s(i,k) = -999.
                enddo
            endif
        enddo
    endif
    if(j.eq.3.OR.j.eq.7.OR.j.eq.11) then
        do i=5,8
            if(mdf(18+i,j).eq.1)then
                do k=1,4
                    ted0s(i,k) = -999.
                enddo
            endif
        enddo
    endif
    if(j.eq.2.OR.j.eq.6.OR.j.eq.10.OR.j.eq.14) then
        do i=1,4
            if(mdf(22+i,j).eq.1)then
                do k=1,4
                    ted30s(i,k) = -999.
                enddo
            endif
        enddo
    endif
    if(j.eq.4.OR.j.eq.8.OR.j.eq.12) then
        do i=5,8
            if(mdf(18+i,j).eq.1)then
                do k=1,4
                    ted30s(i,k) = -999.
                enddo
            endif
        enddo
    endif
    if(j.eq.15) then
        if(mdf(1,j).eq.1) tedback(1,1) = -999.
        if(mdf(23,j).eq.1) tedback(1,2) = -999.
        if(mdf(26,j).eq.1) tedback(1,3) = -999.
        if(mdf(25,j).eq.1) tedback(1,4) = -999.
        if(mdf(24,j).eq.1) tedback(2,2) = -999.
    endif
    if(j.eq.16) then
        if(mdf(1,j).eq.1) tedback(2,1) = -999.
```

```
              if(mdf(2,3).eq.1) tedback(2,3) = -999.
              if(mdf(24,j).eq.1) tedback(2,4) = -999.
          endif
      enddo
* ------------------------------------------------------
*      ! Ted flux
* ------------------------------------------------------
      tf = 1
      do i=1,4
          do n = 1,7
              do j = 1,4
                  k = (i-1)*4 + j
                  tedfx(n,k) = (arc(tf+2096)*b1 +
     +                          arc(tf+2097)*b2 +
     +                          arc(tf+2098)*b3 +
     +                          arc(tf+2099))*cf
                  tf = tf + 4
      !print *,n,k,tf+2096
              enddo
          enddo
      enddo
* ------------------------------------------------------
*      ! Check missing data flags
* ------------------------------------------------------

  100 continue
      return
      end
```