EPA-600/4-75-005-b
May 1975

Environmental Monitoring Series

DEVELOPMENT OF AN URBAN AIR QUALITY SIMULATION MODEL WITH COMPATIBLE RAPS DATA

VOLUME II

U.S. Environmental Protection Agency
Office of Research and Development
Washington, D. C. 20460

EPA-600/4-75-005-b

DEVELOPMENT OF AN URBAN AIR QUALITY SIMULATION MODEL WITH COMPATIBLE RAPS DATA VOLUME II

by

C.C. Shir and L.J. Shieh

IBM Research Laboratory
San Jose, California 95193

Contract No. 68-02-1833
ROAP No. 26AA123
Program Element No. 1AA003

EPA Project Officer: Robert E. Eskridge

Chemistry and Physics Laboratory
Office of Research and Development
Research Triangle Park, N. C. 27711

Prepared for

U.S. ENVIRONMENTAL PROTECTION AGENCY
Office of Research and Development
Washington, D. C. 20460

May 1975

EPA REVIEW NOTICE

This report has been reviewed by the National Environmental Research Center - Research Triangle Park, Office of Research and Development, EPA, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into series. These broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and maximum interface in related fields. These series are:

1. ENVIRONMENTAL HEALTH EFFECTS RESEARCH
2. ENVIRONMENTAL PROTECTION TECHNOLOGY
3. ECOLOGICAL RESEARCH
4. ENVIRONMENTAL MONITORING
5. SOCIOECONOMIC ENVIRONMENTAL STUDIES
6. SCIENTIFIC AND TECHNICAL ASSESSMENT REPORTS
9. MISCELLANEOUS

This report has been assigned to the ENVIRONMENTAL MONITORING series. This series describes research conducted to develop new or improved methods and instrumentation for the identification and quantification of environmental pollutants at the lowest conceivably significant concentrations. It also includes studies to determine the ambient concentrations of pollutants in the environment and/or the variance of pollutants as a function of time or meteorological factors.

This document is available to the public for sale through the National Technical Information Service, Springfield, Virginia 22161.

Publication No. EPA-600/4-75-005-b

DEVELOPMENT OF URBAN AIR QUALITY SIMULATION MODEL WITH COMPATIBLE RAPS DATA

by

C. C. Shir and L. J. Shieh

IBM Reseach Laboratory, San Jose, California 95193

IBM Scientific Center, Palo Alto, California 94304

APPENDIX to

Final Report

Contract No. 68-02-1833

Prepared for
Meteorology Laboratory
U. S. Environmental Protection Agency
Research Triangle Park, N.C. 27711

May 1975

Table of Contents

APPENDIX

1. Main Program Listing

1. IBMAQ-2 (MAIN)
2. AACOMP
3. AKZCAL
4. CADJUS
5. CHEMIC
6. CONSIN
7. DIMENS
8. DTTEST
9. GEOIN
10. OUTAPE
11. POSITV
12. PRINTS
13. SHIFTN
14. SOURCE
15. SOUSIN
16. STABIT
17. STNCON
18. TIMEX
19. UVFLUX
20. UVINTP
21. UVZF
22. WINDER
23. WINDGR
24. WINDIN
25. WRITES
26. WWFLUX
27. WWZF
28. XYDIFF
29. XYUTMS
30. ZZDIFF
31. ZZGRID

```
LEVEL 21.6 ( MAY 72 )                OS/360  FORTRAN H                DATE  75.120/09.05.

COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
C
C          ************************
C          ***  PROGRAM IBMAQ-2  ***
C          ************************
C
C.... THIS PROGRAM COMPUTES 3-D CONCENTRATION DISTRIBUTION IN ST. LOUIS
C....  AREA. REGION IS (40KM X 60KM X HH) IN (30,40,14) GRIDS.
C.... THERE ARE 1200 AREA SOURCES AND 150 POINT SOURCES.
C.... GRADIENT TYPE CONCENTRATION EQUATION IS USED IN THIS MODEL.
C     THE PROGRAM IS AN EXPERIMENTAL PROGRAM. IT IS FOR RESEARCH
C     PURPOSE ONLY.
C     THE MODEL IS DISCUSSED IN IBM RESEARCH REPORT RJ1227,
C     'A GENERALIZED URBAN AIR POLLUTION MODEL AND ITS APPLICATION
C     TO THE STUDY OF SO2 DISTRIBUTION IN ST. LOUIS METROPOLITAN AREA',
C     BY SHIR AND SHIEH, 1973.
C
C ********************************************
C *** SUBROUTINES INCLUDED IN THIS PROGRAM ***
C ********************************************
C
C      NAME   CALLED FROM   FUNCTIONS
C      ------ ------------  ----------------------------------------
C     AACOMP  MAIN          COMPUTE THE CONCENTRATION FIELDS
C     AKZCAL  MAIN          COMPUTE EDDY DIFFUSIVITY
C     CADJUS  MAIN          ADJUST C VALUES DUE TO CHANGE IN GRID DIMEN.
C     CCHECK  AACOMP        CHECK FOR STEADY STATE CONDITION
C     CHEMIC  AACOMP        COMPUTE CHEMICAL DECAY
C     CONSIN  MAIN          SPECIFY MODEL PARAMETERS
C    *CDTOTP  MAIN          PRINT CARD IMAGE OF NAMELIST INPUT
C     DIMENS  MAIN          INITIALIZE GRID SYSTEM
C    *DIMEN1  MAIN          SET VERTICAL GRID SYSTEM
C    *HHCALC  MAIN          COMPUTE TIME VARYING MIXING HEIGHT
C     DTTEST  MAIN          TIME STEP FOR NUMERICAL METHOD
C     GEOIN   MAIN          INPUT GEOGRAPH. AND ANNUAL EMISSION DATA
C     OUTAPE  MAIN          OUTPUT RESULTS TO TAPE OR DISK
C     POSITV  (NOT USED)    SET C=0 IF IT IS LESS THAN ZERO
C     PRINTS  MAIN          PRINT GEOGRAPH. AND ANNUAL EMISSION DATA
C    *PRINTA  MAIN          PRINT TIME VARYING EMISSION RATES
C    *PRINTB  MAIN          PRINT TIME VARYING METEOROLOGICAL DATA
C    *PRINTC  MAIN          PRINT CONCENTRATION FIELDS
C     SHIFTN  MAIN,AACOMP   SHIFT ARRAY A TO ARRAY B
C     SOURCE  AACOMP        ADD NEW SOURCE EMISSION INTO THE SYSTEM
C     SOUSIN  MAIN          INPUT TIME VARYING SOURCE EMISSION RATE
C     STABIT  WINDIN        ESTIMATE CONTINUOUS STABILITY CLASSES
C     STNCON  PRINTC        COMPUTE CONC. VALUES AT RAMS STATIONS
C     TIMEX   MAIN          FIX TIME INDICES
C     UVZF    WINDIN        COMPUTE VERTICAL WIND PROFILE OF (U,V)
C     UVFLUX  AACOMP        COMPUTE HORIZONTAL ADVECTION
C     UVINTP  WINDIN,WINDGR INTERPOLATE ANALYZED U,V TO NUMERICAL GRID
C     WWZF    WINDIN        COMPUTE W COMONENT OF WIND
C     WINDER  WINDGR,UVZF   CONVERT WIND VECTOR TO COMP. OR VICE VERSA
C     WINDGR  WINDIN        GENERATE SFC. WIND FIELD FROM RAMS DATA
C     WINDIN  MAIN          READ IN SURFACE WIND FIELD AND RAMS DATA
C     WRITES  PRINTS        PRINT DATA ARRAY
C    *WRITEX  PRINTS        PRINT HORIZONTAL FIELD OF ARRAY
C    *WRITEZ  PRINTC        PRINT VERTICAL CROSS-SECTION OF ARRAY
```

```
C      WWFLUX  AACOMP         COMPUTE VERTICAL ADVECTION
C      XYDIFF  MAIN           COMPUTE HORIZONTAL DIFFUSION
C      XYUTMS  GEOIN          COMPUTE UTM COORDINATE OF NEMERICAL GRID
C     *XYUTM1  GEOIN,SOURCE   CONVERT (X,Y) FROM UTM TO NUMERICAL GRID
C      ZZDIFF  AACOMP         COMPUTE VERTICAL DIFFUSION
C      ZZGRID  SOURCE         CONVERT Z FROM METER TO NUMERICAL GRID UNIT
C      ------  -------------  ------------------------------------------
C      (NOTE: * DENOTE ENTRY POINT TO LAST STATED SUBROUTINE)
C
C ***************************
C *** TABLE FOR I/O UNITS ***
C ***************************
C
C      UNIT  *    DSNAME    *I/O* ROUTINE *    VARIABLES
C      ------*--------------*---*---------*-----------------------------
C      IUNIT                  I    MAIN      'NAMELIST'  (IUNIT=IUNIT2)
C      IUNIT1                 I    CDTOTP     (UNIT FOR CARD READER)
C      IUNIT2                I/O   CDTOTP     (SCRATCH STORAGE UNIT)
C      JUNIT                  O    (ALL)      (UNIT FOR LINE PRINTER)
C      JUNIT1                 O    CDTOTP     (UNIT FOR LINE PRINTER)
C      KUNITG   EPAGEO2       I    GEOIN     XRAMS,ZS,Z0,QB,PQB,....
C      KUNITS   SO2SOUS1      I    SOUSIN    KHR,KMO,KDAY,KYR,QB,PQB,
C      KUNITW   WINDDATA      I    WINDIN    KYR,KMO,KDAY,KHR,U1,V1,RAMS
C      KUNITW   WINDDATB      I    WINDIN    KYR,KMO,KDAY,KHR,UU,VV,RAMS
C      KUNITW   EPARAMS       I    WINDIN    KYR,KMO,KDAY,KHR,RAMS
C      KUNITC   EPASTNO1      O    OUTAPE    IYR,IMO,IDAY,IHR,PARM,ICAL,IOBS
C      KUNITP   EPACONC1      O    OUTAPE    IYR,IMO,IDAY,IHR,PARM,CC,
C                                            ICAL,IOBS
C      -----------------------------------------------------------------
C
C *************************************************
C *** COMMENTS ON VARIABLES USE IN THIS PROGRAM ***
C *************************************************
C
C .... C .... C .... C .... C
C .. APPEAR IN DIMENSION .. C
C .... C .... C .... C .... C
C
C      CP1,C           = NEW AND OLD CONCENTRATION FIELD (UG/M3)
C      U,V,W           = WIND COMPONENT IN X,Y,Z DIRECTIONS (M/SEC)
C      COLD              OLD SURFACE C FOR CHECKING CONVERGENCE
C      CC              = (HOURLY) AVERAGE SURFACE CONCENTRATION
C      C1              = NEW SURFACE CONCENTRATION
C      U1,V1             SURFACE WIND COMPONENTS IN X,Y DIRECTIONS
C      QA,QB           = NEW,OLD AREA SOURCE EMISSION RATE (G/SEC/KM2)
C                                (CURRENT--ONLY QB IS USED)
C      ZS              = AREA SOURCE HEIGHT (M)
C      Z0              = SURFACE ROUGHNESS (M)
C      AKZ             = VERTICAL EDDY DIFFUSIVITY (M2/SEC)
C      UU,VV             SURFACE WIND FIELD ON WIND GRID POINTS
C      NEAR              NEAREST RAMS STATION TO WIND GRID (IN,JN)
C      DX,DY,DZ          NON-UNIFORM GRID SIZE IN X,Y,Z DIRECTION (M)
C      (DXS,DYS,DZS)     SQUARE OF (DX,DY,DZ)
C      (RDX,RDY,RDZ)     (DX(I)/DX(I-1),DY(J)/DY(J-1),DZ(K)/DZ(K-1))
C      Z                 HEIGHT OF GRID POINT FROM SURFACE (M)
C      ZM                HEIGHT OF POINT IN THE MIDDLE OF GRID (M)
C      AKH               HORIZONTAL EDDY DIFFUSIVITY (M2/SEC)
C      AKF               FUNCTION DETERMINES VERTICAL VARIATION OF AKZ
```

```
C      UZF,VZF,WZF   = FUNCTION DETERMINES VERTICAL VARIATION OF U,V,W
C      NP            = POINT SOURCE IDENTIFICATION NUMBER
C      XP,YP,ZP      = X,Y,Z COORDINATE OF POINT SOURCE
C      ZR            = NORMALIZED PLUME RISE (M*(M/SEC)**.75)
C      ZPR           = EFFECTIVE STACK HEIGHT OF POINT SOURCES (M)
C      PQA,PQB       = NEW, OLD POINT SOURCE EMISSION RATE (G/SEC)
C                          (CURRENT--ONLY PQB IS USED)
C      EK,FK         = TEMPORARY STORAGE ARRAY
C      USTN,VSTN     = U,V COMPONENTS AT RAMS STATION
C      IS,XS,YS      = NUMBER AND X,Y LOCATION OF RAMS STATIONS
C      (JXS,JYS)     = (XS,YS) ON GRID UNIT
C      ICAL,IOBS     = (HOURLY) CAL. AND OBS. C AT RAMS STATION (UG/M3)
C                            (ICAL(NSX),IOBS(NSX) = SPATIAL AVERAGE)
C      KCAL,KOBS     = 24 HOURS AVERAGE OF ICAL, IOBS
C      ITOBS         = NUMBER OF OBS. DATA AT STATION FOR 24 HR. PERIOD
C      IUTM,JUTM     = UTM COORDINATE OF NUMERICAL GRID POINTS
C      XPUTM,YPUTM   = UTM COORDINATES OF POINT SOURCE
C      XSUTM,YSUTM   = UTM COORDINATES OF RAMS STATION
C      AKA           = CHEMICAL REACTION RATE CONSTANT (/SEC)
C      NMONDY        = NUMBER OF DAYS IN A MONTH
C
C .... C .... C .... C ... C
C .. APPEAR IN NAMELIST .. C
C .... C .... C .... C ... C
C
C      IM,JM,KM  = 3-D DIMENSION IN X,Y,Z DIRECTION
C      TM,ITM    = SIMULATED TIME IN SECOND
C      DT        = TIME STEP (SEC)
C      IN,JN     = DIMENSION IN X,Y DIRECTION OF WIND GRID
C      KN        = NUMBER OF LEVELS THAT WIND FIELD  WILL BE COMPUTED
C      KNN       = NUMBER OF FIXED GRID INTERVAL IN VERTICAL DIRECTION
C      LM        = TOTAL NUMBER OF POINT SOURCES
C      NS        = TOTAL NUMBER OF RAMS STATIONS
C      IMC,JMC   = THE CROSS SECTION TO PRINT VERTICAL C DISTRIBUTION
C      HS        = AVERAGED EFFECTIVE HEIGHT OF SURFACE WIND
C      HP        = HEIGHT OF UPPER WIND MEASUREMENT
C      HG        = THICKNESS OF PLANETARY BOUNDARY LAYER
C      HMIN,HMAX = MIN. AND MAX. MIXING HEIGHT OF A DAY
C      ZRPQ      = RATIO OF PLUME RISE TO SOURCE EMISSION RATE
C      ZRISE     = PARAMETER FOR ADJUSTING PLUME RISE
C      PMAX,PMIN = MAX. AND MIN. FOR POWER LAW CONSTANT OF WIND PROFILE
C      DCMIN     = CRITERIA FOR CONVERGENCE OF STEADY STATE
C      OLMIN     = MIN. LIMIT OF OBUKHOV-MONIN LENGTH
C      IHR,IDAY,IMO,IYR = REAL TIME OF HOUR, DAY, MONTH & YEAR
C      LTSTOP    = MAX. NUMBER OF HOURS TO BE SIMULATED IN A RUN
C      LTSOUS    = TIME INTERVAL FOR INPUT SOURCE DATA (SECONDS)
C      LTWIND    = TIME INTERVAL FOR INPUT WIND & METEOR DATA (SECONDS)
C      IDAYTP,IHRTP = STARTING DAY AND HOUR THAT DATA STORED ON
C                          I/O UNIT = KUNITP & KUNITC
C      JUNIT     = OUTPUT UNIT FOR LINE PRINTER
C      KUNITG    = INPUT UNIT FOR GEOGRAPHICAL & NEDS DATA
C      KUNITS    = INPUT UNIT FOR HOURLY SOURCES EMISSION RATE
C      KUNITW    = INPUT UNIT FOR OBJ. WIND & RAMS DATA
C      KUNITW    = INPUT UNIT FOR SUBJ. WIND (UU,VV) & RAMS DATA
C      KUNITW    = INPUT UNIT FOR RAMS DATA
C      KUNITC    = I/O UNIT FOR COMP. SO2 CONC AT STATION
C      KUNITP    = I/O UNIT FOR COMPUTED SURFACE CONC. FOR A TAPE
C      LCRUN     = FLAG TO DECIDE DEBUG RUN OR ACTUAL RUN
```

```
C     LHJUS         FLAG FOR ADJUSTING  C VALUE DUE TO GRID CHANGED
C     LCHEM         FLAG TO DECIDE THE CHEMICAL REACTION COMPUTATION
C     LWW           FLAG FOR COMPUTING VERTICAL WIND ADVECTION
C     LTOP          FLAG FOR CHOICE OF UPPER BOUNDARY CONDITION-DIFFUSION
C     LWTOP         FLAG FOR CHOICE OF UPPER BOUNDARY CONDITION-ADVECTION
C     LSOUS         FLAG USED IN ADDING NEW SOURCE INTO SYSTEM
C     LPQ           FLAG FOR CHOICE OF MODELING POINT SOURCE
C     LWIND         FLAG FOR COMPUTING UPPER LAYER WIND FIELD
C     KWIND         FLAG FOR CHOICE OF OBJECTIVE OR SUBJECTIVE
C                     ANALYZED WIND FIELD
C     Z0MEAN        VALUE THAT UNIFORM SFC ROUGHNESS TO BE USED
C     LWRITE(N)     CONTROL FLAG FOR OUTPUT.
C       ROUTINE     N   COND.    VARIABLES TO BE OUTPUT
C       -------    ---  ------   ------------------------------------
C       PRINTS      1   .GE. 1   IS,XSUTM,YSUTM,XS,YS,JXS,JYS(L),L=1,NS
C                                Z0(I,J),ZS(I,J),QB(I,J),QBTOT,QBSUM
C                                NP,XPUTM.YPUTM,PQB,XP,YP,ZP,ZR(L),L=1,LM
C                                PQBTOT,PQBSUM
C       PRINTA      2   .GE. 1   QBTOT,PQBTOT,QBSUM,PQBSUM
C                          2,4   QB(I,J)
C                       = 3,4    PQB(L)
C       PRINTB      3   .GE. 1   RAMS(M,L)
C                       =    2   UU(I,J),VV(I,J)
C                   4   .GE. 2   U1(I,J),V1(I,J)
C                                + W(I,J,1), (IF LWW=1)
C                            3   U(I,J,K),V(I,J,K)
C                                + W(I,J,K), (IF LWW=1)
C                       .NE. 0   Z(K),RH
C                       .GE. 1   UZF(K),VZF(K)
C                                + WZF(K), (IF LWW=1)
C                   5   .GE. 1   AKF(K)
C                            2   AKZ(I,J), FOR J=JM/2
C                            3   AKZ(I,J)
C                   6   .GE. 1   PARM(L),DT
C                       =    2   U0,PHIFHZ,HFZ,RIB
C       PRINTC      7      2,4   CP1(JXS,JYS)
C                       .GE. 1   CP1(IMC,J,K),CP1(I,JMC,K)
C                       .GE. 3   CP1(I,J,1)
C                                CP1(I,J,K), (IF IHR=0)
C                   8   .GE. 1   ICAL(L),IOBS(L)
C                                + KCAL(L),KOBS(L), (IF IHR=0)
C                       =    2   CC(I,J)
C       OUTAPE      9   .GE. 1   IYR,IMO,IDAY,IHR,PARM(M),ICAL(L),IOBS(L)
C                                          -- ON I/O UNIT=KUNITC
C                  10   .GE. 1   IYR,IMO,IDAY,IHR,PARM(M),CC(I,J),ICAL(L),
C                                & IOBS(L) -- ON I/O UNIT=KUNITP
C       -----------------------------------------------------------
C
C ... C .... C .... C ... C
C ... APPEAR IN COMMON .. C
C ... C .... C .... C ... C
C
C     IM1,JM1,KM1     IM,JM,KM MINUS ONE
C     ITM,TM        = SIMULATED TIME IN SECONDS
C     ITSEC,TSEC    = TIME IN SECONDS STARTING FROM EACH 1 HR INTERVAL
C     ITOTHR          TOTAL REAL TIME BEING SIMULATED IN HOURS
C     ITMHR         = SIMULATED TIME IN HOUR ( = ITM/3600*3600)
C     ITSTEP          NUMBER OF TIME STEPS
```

```
          C      LPRINT       = MIN. OF (LTWIND,LTSOUS), TIME INTERVAL FOR
          C                     TAKING AVERAGE OF CONCENTRATION
          C      RAMS(M,N)    = RAMS STATION DATA, N -- STATION INDEX
          C                    M=1 WIND SPEED;        =2 WIND DIRECTION;
          C                     =3 1ST LEVEL TEMP.;   =4 2ND LEVEL TEMP.;
          C                     =5 SO2 CONCENTRATION; =6 RADIATION
          C      PARM(M)      = METEOROLOGICAL PARAMETERS
          C                    M=1 MEAN WIND SPEED;     =2 WIND DIRECTION;
          C                     =3 SURFACE TEMP.;       =4 STABILITY CLASS;
          C                     =5 MIXING HEIGHT;       =6 AREA SOURCE Q;
          C                     =7 POINT SOURCE Q;      =8 UPPER TEMP.;
          C                     =9 RADIATION;           =10 OBUKHOV LENGTH
          C      A1           = CONSTANTS IN BUSINGER'S FORMULA
          C      AK           = VON KARMEN CONSTANT
          C      RIB          = BULK RICHARDSON NUMBER
          C      ZMAX         = MAXIMUN  MIXING HEIGHT USED IN THE MODEL
          C      QBTOT,PQBTOT = TOTAL AREA AND POINT SOURCES Q OF INPUT DATA
          C      U0           = FRICTION VELOCITY
          C      PHIFHZ       = NON-DIMENSIONAL TEMP. GRADIENT
          C      HFZ          = INTEGRAL OF NON-DIMENSIONAL WIND SHEAR
          C
          C
          C *********************
          C ***  MAIN PROGRAM ***
          C *********************
          C
ISN 0002         DIMENSION
          C     *     CP1(30,40,14),C(30,40,14),U(30,40,7),V(30,40,7),W(30,40,7)
          C     *     CP1(30,40,14),C(30,40,14),U(30,40,1),V(30,40,1),W(30,40,1)
          C     *     CP1(30,40,5),C(30,40,5),U(30,40,1),V(30,40,1),W(30,40,1)
          C     *     U(30,40,7),V(30,40,7),W(30,40,7)
                *     U(30,40,1),V(30,40,1),W(30,40,1)
                *     ,COLD(30,40),  CC(30,40),  C1(30,40),  U1(30,40),  V1(30,40)
                *     ,  QA(30,40),  QB(30,40),  ZS(30,40),  Z0(30,40), AKZ(30,40)
                *     ,  UU( 9,13),  VV( 9,13),NEAR( 9,13)
                *     ,   DX(30),  RDX(30),  DXS(30),   DY(40),  RDY(40),  DYS(40)
                *     ,   DZ(14),  RDZ(14),  DZS(14),    Z(14),    ZM(14),  AKH(14)
                *     ,  AKF(14),  UZF(14),  VZF(14),  WZF(14)
                *     ,  NP(150),  XP(150),  YP(150),  ZP(150),  ZR(150), ZPR(150)
                *     , PQA(150), PQB(150),  EK(150),  FK(150), USTN(25), VSTN(25)
                *     ,    IS(25),   XS(25),   YS(25),  JXS(25),  JYS(25)
                *     , IOBS(26), ICAL(26), KOBS(26), KCAL(26),ITOBS(26)
                *     ,IUTM(30),JUTM(40),XPUTM(150),YPUTM(150),XSUTM(25),YSUTM(25)
                *     , AKA(24), NMONDY(12)
          C
ISN 0003         COMMON   /AADATA/
                *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,OCMIN
                *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
ISN 0004         COMMON /CBLOCK/ CP1(30,40,14),C(30,40,14)
          C      COMMON /CBLOCK/ CP1(30,40,5),C(30,40,5)
          C
ISN 0005         EQUIVALENCE
                *        (CP1(1,1,1), C1(1,1))
                *      ,(  U(1,1,1),  W(1,1,1))
```

```
                *       ,(U(1,1,1),U1(1,1)), (V(1,1,1),V1(1,1))
                *       , (QA(1,1),QB(1,1)),    (PQA(1), PQB(1))
                *       ,   (UZF(1),VZF(1)),     (UZF(1),WZF(1))
            C   *       ,   (UU(1,1),EK(1)),     (VV(1,1),FK(1)), (NEAR(1,1),NP(1))
            C   *       , (USTN(1),EK(120)),  (VSTN(1),FK(120))
            C
ISN 0006          NAMELIST   /INLIST/
                *    IM,JM,KM,DX,DY,DZ,TM,DT,IN,JN,KN,KNN,LM,NS,IMC,JMC, AKH
                *   ,AKA,HS,HP,HG,ZMAX,HMIN,HMAX,ZRPQ,ZRISE,PMAX,PMIN,DCMIN,OLMIN
                *   ,IHR,IDAY,IMO,IYR,LTSTOP,LTSOUS,LTWIND,IDAYTP,IHRTP
                *   ,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                *   ,LCRUN,LHJUS,LCHEM,LWW,LTOP,LWTOP,LSOUS,LPQ,KWIND,LWIND,LWRITE
                *   ,ZOMEAN
            C
ISN 0007          DATA
                *    ZS/1200*10./,Z0/1200*1./,QA/1200*0./,CC/1200*0./,COLD/1200*0./
                *    ,RH/1.0/,ZPR/150*20.0/
                *    ,UZF/14*1./,VZF/14*1./,WZF/14*1./,AKF/14*1./
                *    ,IOBS/26*0/,ICAL/26*0/,KOBS/26*0/,KCAL/26*0/,ITOBS/26*0/
                *    ,NMONDY/31,28,31,30,31,30,31,31,30,31,30,31/
ISN 0008          DATA IUNIT1/5/,IUNIT2/12/,JUNIT1/6/
            C
            C **************
            C ***  INPUT ***
            C **************
            C
            C *** COPY INPUT CARDS TO DISK TO BE LISTED ***
            C
ISN 0009          CALL      CDTOTP      (JUNIT1,IUNIT1,IUNIT2)
ISN 0010          IUNIT=IUNIT2
ISN 0011          READ  (IUNIT,INLIST)
            C     WRITE (JUNIT,INLIST)
            C
            C *** INPUT CONSTANTS ***
            C
ISN 0012          CALL      CONSIN      (IM,JM,KM,KN,IMJM,IJKM,IJKN)
            C
            C *** INITIALIZE ARRAY ***
            C
ISN 0013          DO 50 J=1,JM
ISN 0014          DO 50 I=1,IM
ISN 0015          DO 20 K=1,KN
ISN 0016          U(I,J,K)=0.0
ISN 0017          V(I,J,K)=0.0
ISN 0018          W(I,J,K)=0.0
ISN 0019       20 CONTINUE
ISN 0020          DO 30 K=1,KM
ISN 0021          CP1(I,J,K)=0.0
ISN 0022          C   (I,J,K)=0.0
ISN 0023       30 CONTINUE
ISN 0024       50 CONTINUE
            C
            C *** INPUT GEOGRAPHIC AND NEDS DATA ***
            C
ISN 0025          LMAX=LM
ISN 0026          CALL      GEOIN
                & (QA,ZS,Z0,PQA,XPUTM,YPUTM,XP,YP,ZP,ZR,NP,IUTM,JUTM,DX,DY,IM,JM
                &    ,LMAX,LM,XSUTM,YSUTM,XS,YS,IS,JXS,JYS,25,NS,KUNITG,JUNIT,QBSUM
```

```
                 &    ,PQBSUM,QBTOT,PQBTOT,ZOMEAN)
              C
              C.... QBSUM  = TOTAL AREA SOURCE EMISSION WITHIN THE COMPUTING REGION
              C                OR TOTAL EMISSION USED IN THE MODEL.
              C      PQBSUM = SAME DEF. AS QBSUM FOR POINT SOURCE
              C      QBTOT  = TOTAL AREA SOURCE EMISSION FROM INPUT DATA
              C      PQBTOT = TOTAL POINT SOURCE EMISSION FROM INPUT DATA
ISN 0027             NSX=NS+1
ISN 0028             PARM(6)=QBSUM
ISN 0029             PARM(7)=PQBSUM
              C
              C *** INITIALIZE NUMERICAL GRID SYSTEM ***
              C
ISN 0030             CALL      DIMENS  (DX,DY,DZ,RDX,RDY,RDZ,DXS,DYS,DZS,Z,ZM,IM,JM,KM)
              C
              C *** PRINT GEOGRAPHIC AND NEDS DATA ***
              C
ISN 0031             CALL      PRINTS
                   & (CP1,CC,C1,COLD,IOBS,ICAL,KOBS,KCAL,ITOBS,U,V,W,U1,V1,UU,VV
                   & ,AKZ,UZF,VZF,WZF,AKF,DX,DY,IUTM,JUTM,XSUTM,YSUTM,XPUTM,YPUTM
                   & ,ZO,ZS,QB,PQB,XP,YP,ZP,ZR,NP,IN,JN,KN,LM,IS,XS,YS,JXS,JYS,Z
                   & ,IM,JM,KM,NS,NSX)
              C
              C **************************
              C *** ENTERING MAIN LOOP ***
              C **************************
              C
              C      IF (ICASE .EQ. 2) GO TO 2000
              C
ISN 0032        1000 CONTINUE
              C
              C *** SEARCH AND STORED DATA OF CC,ICAL,IOBS ***
              C.... SUBROUTINE OUTAPE PERFORM I/O ON UNIT=KUNITP & KUNITC.
              C
ISN 0033             CALL       OUTAPE      (CC,ICAL,IOBS,NSX,IM,JM,IDAYTP,IHRTP,LWARM)
              C
              C *** TERMINATE THE COMPUTATION IF SPECIFIED TIME BEING REACHED ***
              C
ISN 0034             IF (ITOTHR .GE. LTSTOP) GO TO 9000
              C
              C.... PRINT REAL TIME FOR LINE PRINTER OUTPUT.
ISN 0036             IF (ITM.EQ.ITMHR) WRITE (JUNIT,8000) IYR,IMO,IDAY,IHR
ISN 0038        8000 FORMAT (1H1,' ... IYR,IMO,IDAY,IHR   ',4(I2,'/'),' ......')
              C
              C *** INPUT TIME VARYING SOURCE EMISSION RATE ***
              C
              C      IF (ITM .NE.(ITM/LTSOUS*LTSOUS)) GO TO 100
              C
              C      CALL      SOUSIN
              C     & (QB,ZS,ZO,PQB,XP,YP,ZP,ZR,IM,JM,LM)
              C
              C *** PRINT TIME VARYING SOURCE EMISSION DATA ***
              C
              C      CALL      PRINTA
              C
              C 100 CONTINUE
              C
ISN 0039             IF (ITM.NE.ITMHR) GO TO 200
```

```
          C
          C *** OBTAIN MIXING HEIGHT ***
          C
ISN 0041        CALL     HHCALC       (HH,HMIN,HMAX)
          C.... HH   HOURLY AVERAGE MIXING HEIGHT.
ISN 0042        PARM(5)=HH
          C
          C *** ADJUST VERTICAL GRID DUE TO CHANGE IN MIXING HEIGHT ***
ISN 0043        IF (KNN .GT. KM) KNN=KM
          C
ISN 0045        CALL     DIMEN1       (DZ,RDZ,DZS,Z,ZM,KM,KNN,RH)
          C
ISN 0046        IF((ABS(RH-1.0)) .LE. 0.01) GO TO 200
          C *** ADJUST C DUE TO CHANGE IN VERTICAL GRID ***
          C
ISN 0048        CALL     SHIFTN      (CP1,  C,IM,JM,KM,IM,JM,KM)
          C
ISN 0049        CALL     CADJUS
               & (CP1,C,RDZ,IM,JM,KM,IMJM,IJKM,KNN,RH,LHJUS)
          C
ISN 0050    200 CONTINUE
          C
          C *** READ IN WIND DATA AND CALCULATE HORI & VERT WIND PROFILES ***
ISN 0051        IF (ITM .NE.(ITM/LTWIND*LTWIND)) GO TO 250
          C
ISN 0053        CALL     WINDIN
               & (U,V,W,U1,V1,UU,VV,NEAR,USTN,VSTN,UZF,VZF,WZF,XSUTM,YSUTM
               & ,IUTM,JUTM,Z,DX,DY,DZ,IM,JM,KM,IMJM,IJKN,IN,JN,KN,IS,NS)
          C
          C *** COMPUTE VERTICAL EDDY DIFFUSIVITY ***
          C
ISN 0054        CALL     AKZCAL       (AKZ,U,V,ZO,AKF,AKH,ZM,IMJM,IJKN,KM)
ISN 0055    250 CONTINUE
          C
          C *** CHECK TIME STEP ***
ISN 0056        IF(ITM .NE.(ITM/LTWIND*LTWIND)) GO TO 300
ISN 0058        AKAHR=AKA(IHR+1)
          C
ISN 0059        CALL     DTTEST
               & (U,V,AKH,AKZ,AKF,AKAHR,UZF,VZF,DX,DY,Z,IM,JM,KM,IMJM,IJKN,KN)
          C
          C *** PRINT METEOROLOGICAL DATA AND PARAMETERS OF NUMERICAL GRID ***
          C
ISN 0060        CALL     PRINTB      (RH)
          C
          C ******************************
          C *** ENTERING TIME STEP LOOP ***
          C ******************************
          C
ISN 0061    300 CONTINUE
          C
          C *** COMPUTE CONCENTRATION FIELD ***
          C
ISN 0062        IF (LCRUN.EQ.1)      CALL      AACOMP
               & (CP1,C,CC,COLD,U,V,W,ZS,QA,PQA,AKZ,AKH,AKA,AKF,UZF,VZF,WZF
               & ,DX,RDX,DXS,DY,RDY,DYS,EK,FK,DZ,RDZ,DZS,Z,ZM,XPUTM,YPUTM,XP,YP
               & ,ZP,ZR,ZPR,IM,JM,KM,KN,LM,IMJM,IJKM,IJKN)
          C
```

```
              C *** FIX TIME INDICIES ***
              C
   ISN 0064          CALL      TIMEX      (NMONDY)
              C
              C *** PRINT COMPUTED CONCENTRATION FIELD ***
              C
   ISN 0065          CALL      PRINTC     (IMC,JMC)
   ISN 0066          IF (ITM.NE.(ITM/LPRINT*LPRINT)) GO TO 300
              C
   ISN 0068          GO TO 1000
              C
              C **************************************************
              C *** PROGRAM FOR CASE STUDY AND CLIMATICAL STUDY ***
              C **************************************************
              C.... THIS PORTION OF THE PROGRAM YET TO BE DEVELOPED
              C
              C2000 CONTINUE
              C
              C      CALL      CASE2
              C     & (CP1,C,CC,COLD,U,V,W,ZS,QA,PQA,AKZ,AKH,AKA,AKF,UZF,VZF,WZF
              C     & ,DX,RDX,DXS,DY,RDY,DYS,EK,FK,DZ,RDZ,DZS,Z,ZM,XPUTM,YPUTM,XP,YP
              C     & ,ZP,ZR,ZPR,IM,JM,KM,KN,LM,IMJM,IJKM,IJKN)
              C
   ISN 0069    9000 CONTINUE
   ISN 0070          STOP
              C      DEBUG SUBCHK
   ISN 0071          END
*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       70 ,PROGRAM SIZE =     52948

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              101K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=000^K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE AACOMP
                     & (CP1,C,CC,COLD,U,V,W,ZS,QA,PQA,AKZ,AKH,AKA,AKF,UZF,VZF,WZF
                     & ,DX,RDX,DXS,DY,RDY,DYS,EK,FK,DZ,RDZ,DZS,Z,ZM,XPUTM,YPUTM,XP,YP,ZP
                     &     ,ZR,ZPR,IM,JM,KM,KN,LM,IMJM,IJKM,IJKN)
                C
                C.... THIS ROUTINE COMPUTES THE CONCENTRATION FIELDS.
                C
  ISN 0003            DIMENSION CP1(IM,JM,KM), C(IM,JM,KM),CC(IM,JM),COLD(IM,JM)
                     *    ,U(IM,JM,KN),V(IM,JM,KN),W(IM,JM,KN)
                     *    ,ZS(IM,JM),QA(IM,JM),AKZ(IM,JM)
                     *    ,DX(IM),RDX(IM),DXS(IM),DY(JM),RDY(JM),DYS(JM),EK(LM),FK(LM)
                     *    ,DZ(KM),RDZ(KM),DZS(KM),Z(KM),ZM(KM),AKH(KM)
                     &    ,AKF(KM),UZF(KM),VZF(KM),WZF(KM), XPUTM(LM),YPUTM(LM)
                     *    ,XP(LM),YP(LM),ZP(LM),PQA(LM),ZR(LM),ZPR(LM),AKA(24)
                C
  ISN 0004            COMMON    /AADATA/
                     *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                     *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                     *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                     *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                     *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                     *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                C
                C *** STORE OLD SURFACE C ***
                C
  ISN 0005            CALL        SHIFTN      (CP1,COLD,IM,JM,KM,IM,JM, 1)
                C
                C ***  ADD SOURCES ***
                C
  ISN 0006            CALL        SHIFTN      (CP1,    C,IM,JM,KM,IM,JM,KM)
                C
  ISN 0007            CALL        SOURCE
                     & (CP1, C,QA,QA,ZS,PQA,PQA,XPUTM,YPUTM,XP,YP,ZP,ZR,ZPR,U,V,UZF,VZF
                     & ,AKF,AKH,AKZ,DX,DY,DZ,Z,EK,FK,IM,JM,KM,KN,IMJM,LM)
                C
                C *** CALCULATE HORIZONTAL ADVECTION ***
                C
  ISN 0008            CALL        SHIFTN      (CP1,    C,IM,JM,KM,IM,JM,KM)
                C
  ISN 0009            CALL        UVFLUX
                     & (CP1, C,U,UZF,DX,DY,RDX,RDY,Z,EK,FK
                     & ,IM,JM,KM,IMJM,IJKM, 1,IM,IJKN)
                C
  ISN 0010            CALL        SHIFTN      (CP1,    C,IM,JM,KM,IM,JM,KM)
                C
  ISN 0011            CALL        UVFLUX
                     & (CP1, C,V,VZF,DY,DX,RDY,RDX,Z,EK,FK
                     & ,JM,IM,KM,IMJM,IJKM,IM, 1,IJKN)
                C
                C *** CALCULATE VERTICAL ADVECTION ***
                C
  ISN 0012            CALL        SHIFTN      (CP1,    C,IM,JM,KM,IM,JM,KM)
                C
  ISN 0013            IF (LWW .NE. 0)      CALL        WWFLUX
                     & (CP1, C,W,WZF. Z,DX,DY,DZ,EK,FK,RDX,RDY,RDZ
```

```
                     & ,IM,JM,KM,IMJM,IJKM,IJKN)
              C
              C *** CALCULATE HORIZONTAL DIFFUSION ***
              C
  ISN 0015          IF (LWW.NE.0)     CALL       SHIFTN     (CP1, C,IM,JM,KM,IM,JM,KM)
              C
  ISN 0017          CALL         XYDIFF
                   & (CP1,  C,AKH,AKZ,AKF,RDX,RDY,DXS,DYS,IM,JM,KM,IMJM,IJKM)
              C
              C *** CALCULATE VERTICAL DIFFUSION ***
              C
  ISN 0018          CALL        SHIFTN     (CP1,    C,IM,JM,KM,IM,JM,KM)
              C
  ISN 0019          CALL        ZZDIFF
                   & (CP1, C,AKZ,AKF,RDZ,DZS,ZM,EK,FK,IM,JM,KM,IMJM,IJKM)
              C
              C *** CALCULATE CHEMICAL DECAY ***
              C
  ISN 0020          CALL        SHIFTN     (CP1,    C,IM,JM,KM,IM,JM,KM)
              C
  ISN 0021          IF (LCHEM .NE. 0)     CALL       CHEMIC
                   & (CP1,C,IM,JM,KM,IMJM,IJKM,AKA)
              C
  ISN 0023          IF (LCHEM.NE.0)     CALL      SHIFTN     (CP1,C,IM,JM,KM,IM,JM,KM)
              C
  ISN 0025          IF  (ITSEC .LT. 900) GO TO 100
              C
              C *** CHECK IF STEADY STATE SOLUTION HAVE BEEN OBTAINED ***
              C
  ISN 0027          ICHECK=0
              C
  ISN 0028          CALL        CCHECK     (CP1,COLD,IJKM,IMJM,ICHECK,DCMIN)
              C
  ISN 0029          IF (ICHECK .EQ. 1) GO TO 100
              C
              C...  STEADY STATE REACHED .....
  ISN 0031          DT=LPRINT-TSEC
  ISN 0032      100 CONTINUE
              C
              C *** TAKE TIME AVERAGE CC ***
              C
  ISN 0033          DTPRNT=DT/LPRINT
  ISN 0034          DO 200 J=1,JM
  ISN 0035          DO 200 I=1,IM
  ISN 0036          CC(I,J)=CC(I,J)+CP1(I,J,1)*DTPRNT
  ISN 0037      200 CONTINUE
              C
  ISN 0038          RETURN
              C     DEBUG SUBCHK
  ISN 0039          END
*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      38 ,PROGRAM SIZE =     3162

*STATISTICS*  NO  DIAGNOSTICS GENERATED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE AKZCAL (AKZ, U, V, ZO,AKF,AKH,ZM,IMJM,IJKN,KM)
                  C
                  C.... IT ESTIMATES THE EDDY AKZ FROM SURFACE WIND SPEED & OBUKHOV-MONIN
                  C     LENTH WHICH IS CALCULATED FROM THE STABILITY INDEX AND ROUGHNESS
                  C     BY THE FOLLOWING FORMULAS:
                  C     1) 1/L=SIGN(S)*10**(-4/(1+1.3*S**0.85))/((0.216586*LOG(1.2+10/ZO))
                  C            **2).
                  C     2) U0=UV*AK/HFZ, HFZ=INTEG(NON-DIM. WIND SHEAR).
                  C     3) AKZ(Z)= U0*AK*Z/PHIFHZ, PHIFHZ=NON-DIM. TEMP GRADIENT
                  C     AKZ      EDDY DIFFUSIVITY
                  C     U,V      SURFACE WIND
                  C     ZO       SURFACE ROUGHNESS
                  C     OLMIN    MIN. OF OBUKHOV LENGTH
                  C     HH       HEIGHT OF MIXING LAYER
                  C     OL       OBUKHOV-MONIN KENGTH
                  C     S        STABILITY INDEX
                  C
  ISN 0003              DIMENSION    AKZ(IMJM),U(IJKN),V(IJKN),ZO(IMJM),AKF(KM),ZM(KM)
                       *             ,AKH(KM)
                  C
  ISN 0004              COMMON    /AADATA/
                       *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                       *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                       *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                       *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                       *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                       *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                  C
  ISN 0005              DATA PAI/3.14159/
                  C
                  C *** COMPUTE BULK RICHARDSON NUMBER ***
                  C      (CURRENT-- FOR REFERENCE ONLY)
  ISN 0006              RIB=0.0
                  C     TAVE=0.5*(PARM(3)+PARM(8)+273.16
                  C     GZT=9.8016*HS*HS/TAVE
                  C     DTDZ=(PARM(8)-PARM(3)/(30.-HS)+0.00976
                  C     RIB=GZT*DTDZ/(PARM(1)*PARM(1))
                  C
  ISN 0007              S=PARM(4)
  ISN 0008              SIGNS=1.0
  ISN 0009              IF ( S .LT. 0.0) SIGNS=-1.0
  ISN 0011              SS=  ABS(S)
  ISN 0012              O1=10.0**(-4.0/(1.0+1.3*SS**0.85))
  ISN 0013              Z=HS
  ISN 0014              DO 200 I=1,IMJM
  ISN 0015              Z00=ZO(I)
  ISN 0016              OLI=O1*(0.216586* ALOG(1.2+10./Z00))**2
  ISN 0017              OL=SIGNS/OLI
                  C.... MIN. OF OL IS SPECIFIED TO AVOID TOO SMALL VALUES OF OL
  ISN 0018              IF ( ABS(OL) .LT. OLMIN) OL= SIGN (OLMIN,OL)
  ISN 0020              UV= SQRT( U(I)**2+V(I)**2)
  ISN 0021              ZOL=Z/OL
  ISN 0022              AZOL=A1(1)*ZOL
                  C
                  C *** BUSINGER'S FOPMULAS TO ESTIMATE EDDY AKZ ****
```

```
              C
   ISN 0023          IF (OL .LT. 0.0) GO TO 100
              C
              C.... STABLE CASE ....
   ISN 0025          PHIFZ=1.0+AZOL
   ISN 0026          PHIFHZ=A1(4)+AZOL
   ISN 0027          HFZ= ALOG(1.0+Z/Z00)+AZOL
   ISN 0028          GO TO 120
              C
              C.... UNSTABLE CASE ......
   ISN 0029      100 CONTINUE
   ISN 0030          PHIFZ=(1.0-A1(2)*ZOL)**(-0.25)
   ISN 0031          PHIFHZ= A1(4)*(1.0-A1(3)*ZOL)**(-0.5)
   ISN 0032          X=1.0/PHIFZ
   ISN 0033          XA= 2.0*ALOG(0.5*(1.0+X))
   ISN 0034          XB= ALOG(0.5*(1.0+X*X))
   ISN 0035          XC= 2.0*ATAN( X)
   ISN 0036          HFZ= ALOG(1.0+Z/Z00)-XA-XB+XC-0.5*PAI
   ISN 0037      120 CONTINUE
   ISN 0038          U0=UV*AK/HFZ
   ISN 0039          AKZ(I)= U0*AK*Z/PHIFHZ
   ISN 0040      200 CONTINUE
              C
   ISN 0041          PARM(10)=OL
              C
              C *** COMPUTE K AT HIGHER HEIGHT ***
              C
              C      AKZF=K(Z)/K(HS) IS THE RATIO OF TWO K AT TWO HEIGHTS WHICH
              C      IS SUBJECTED TO IMPROVEMENT BY INCORPORATING THE INTERNAL
              C      TURBULENCE BOUNDRY LAYER DUE TO HORIZONTAL INHOMOGENEITY....
              C
   ISN 0042          HH=PARM(5)
              C
   ISN 0043          DO 400 K=1,KM
              C.... LIMIT THE HEIGHT BETWEEN ZMAX AND HS.
   ISN 0044          ZZ=ZM(K)
   ISN 0045          IF ( ZM(K).LT. 20) ZZ=20.
   ISN 0047          IF (ZM(K).GT. ZMAX) ZZ=ZMAX
   ISN 0049          AKZF=ZZ/HS*EXP(-0.004*(ZZ-HS))
              C      AKZF=0.5 WHEN ZM(K) IS ABOVE MIXING HEIGHT OR AKZF IS TOO SMALL.
   ISN 0050          IF ((ZM(K).GT. HH) .OR.   (AKZF .LT. 0.5)) AKZF=0.5
   ISN 0052          AKF(K)=AKZF
   ISN 0053      400 CONTINUE
              C
   ISN 0054          RETURN
              C      DEBUG SUBCHK
   ISN 0055          END

*OPTIONS IN EFFECT*     NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*     SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*    SOURCE STATEMENTS =      54 ,PROGRAM SIZE =     1670

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                  OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE CADJUS (CP1,C,RDZ,IM,JM,KM,IMJM,IJKM,KNN,RH,LHJUS)
                  C
                  C.... THIS SUBROUTINE ADJUST THE CONCENTRATION VALUES DUE TO CHANGE
                  C      IN VOLUME OF GRID CELLS.
                  C.... RH   = RATIO OF OLD TO NEW VERTICAL GRID SIZE
                  C     IRH  = MIN. OF (RH+1,2), =1, IF (RH.LE.1); =2, IF (RH.GT.1)
                  C     LHJUS= CONTROL FLAG TO DECIDE WHICH METHOD TO ADJUST THE CONC.
                  C             VALUE DUE TO GRID CHANGE
                  C          ---------------------- METHOD -------------------------
                  C             LHJUS =        0        1         2         3
                  C     IRH = 1                A        A         C         C
                  C     IRH = 2                A        B         A         B
                  C                      -----------------------------
                  C     METHOD A : KEEP THE TOTAL MASS CONSTANT
                  C     METHOD B : STRETCH GRID SPACE BUT KEEP OLD CONC. VALUE
                  C                  (USED ONLY WHEN RH.GE.1 -- RISING MIXING HEIGHT)
                  C     METHOD C : COMPRESS GRID SPACE WHILE OBTAIN THE NEW CONC. VALUE
                  C                 AT CORRESPONDING HEIGHT OF OLD CONC. PROFILE
                  C                  (USED ONLY WHEN RH.LT.1 -- LOWERING MIXING HEIGHT)
                  C
  ISN 0003              DIMENSION CP1(IJKM),C(IJKM),RDZ(KM),LHH(2)
                  C
  ISN 0004              IF ( RH .LE. 0.0) RETURN
                  C
                  C.... FIX CONTROL PARAMETERS
  ISN 0006              LH=LHJUS
  ISN 0007              IRH=RH+1
  ISN 0008              IF ( IRH .GT. 2) IRH=2
  ISN 0010              LHH(1)=LH/2
  ISN 0011              LHH(2)=LH-(LH/2)*2
  ISN 0012              IF ( LHH(IRH) .EQ. 1) GO TO 200
                  C
                  C *** DILUTION (METHOD A) ***
                  C.... RFRH IS DILUTION FACTOR.
  ISN 0014              KK=(KNN-2)*IMJM
  ISN 0015              RF=(RH+RDZ(KNN))/(1.0+RDZ(KNN))
  ISN 0016              DO 100 K=KNN,KM
  ISN 0017              IF ( K .GT. KNN) RF=1.0
  ISN 0019              RFRH=RF/RH
  ISN 0020              KK=KK+IMJM
  ISN 0021              DO 100 I=1,IMJM
  ISN 0022              IJK=I+KK
  ISN 0023              CP1(IJK)= C(IJK) *RFRH
  ISN 0024          100 CONTINUE
                  C
  ISN 0025              RETURN
                  C
                  C *** MOVE GRID POINTS (METHOD B & C) ***
                  C.... METHOD B -- RETURN TO MAIN PROGRAM
                  C       THIS METHOD IS AUTOMATIC WHEN ROUTINE DIMEN1 IS CALLED
  ISN 0026          200 CONTINUE
  ISN 0027              IF ( RH .GE. 1.0) RETURN
                  C
                  C.... METHOD C -- THIS PART OF PROGRAM IS REACHED ONLY WHEN LHH=1
                  C                    AND RH.LT.1 (GRID POINT IS COMPRESSED)
```

```
ISN 0029          KNN1=KNN+1
ISN 0030          KK=IJKM
ISN 0031          DO 300 L=KNN1,KM
ISN 0032          KK=KK-IMJM
ISN 0033          K=KM-L+KNN1
ISN 0034          RHK= ( K-KNN)*RH
ISN 0035          KRH=RHK
ISN 0036          RR=1.0-RHK+KRH
ISN 0037          KKK=(KNN+KRH)*IMJM
ISN 0038          DO 300 I=1,IMJM
ISN 0039          IJKK=KKK+I
ISN 0040          IJK=I+KK
ISN 0041          CP1(IJK)= (1.0-RR)*C(IJKK)+RR*C(IJKK-IMJM)
ISN 0042      300 CONTINUE
            C
ISN 0043          RETURN
            C     DEBUG SUBCHK
ISN 0044          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      43 ,PROGRAM SIZE =      1016

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                         117K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                    OS/360  FORTRAN H

        COMPILER OPTIONS   NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
               C
  ISN 0002           SUBROUTINE CCHECK ( CP1,COLD,IJKM,IMJM,ICHECK,DCMIN)
               C
               C.... THIS SUBROUTINE CHECKS IF CONCENTRATION REACH THE STEADY STATE
               C....  VALUE. IF YES, ICHECK=0, OTHERWISE ICHECK=1.
               C.... DCMIN IS A SPECIFIED NUMBER INPUT BY NAMELIST/INLIST/
               C
  ISN 0003           DIMENSION CP1(IJKM),COLD(IMJM)
               C
  ISN 0004           DO 100 I=1,IMJM
  ISN 0005           DC= CP1(I)-COLD(I)
  ISN 0006           IF (ABS(DC) .LE. DCMIN) GO TO 100
  ISN 0008           ICHECK=1
  ISN 0009       100 CONTINUE
               C
  ISN 0010           RETURN
               C     DEBUG SUBCHK
  ISN 0011           END

*OPTIONS IN EFFECT*     NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*     SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS        10 ,PROGRAM SIZE        380

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                        125K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                OS/360  FORTRAN H

           COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                              SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
   ISN 0002             SUBROUTINE CHEMIC (CP1,C,IM,JM,KM,IMJM,IJKM,AKA)
                  C
                  C.... THIS SUBROUTINE COMPUTES CHEMICAL DECAY OF SO2.
                  C.... AKA IS REACTION RATE CONSTANT.
                  C
   ISN 0003             DIMENSION CP1(IJKM), C(IJKM),  AKA(24)
                  C
   ISN 0004             COMMON    /AADATA/
                       *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                       *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                       *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                       *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                       *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                       *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                  C
   ISN 0005             IHR1=IHR+1
   ISN 0006             IF ( IHR1 .GT. 24) IHR1=1
   ISN 0008             DTAKA=DT*AKA(IHR1)
   ISN 0009             KK=-IMJM
   ISN 0010             DO 100 K=1,KM
   ISN 0011             KK=KK+IMJM
   ISN 0012             JJ=-IM
   ISN 0013             DO 100 J=1,JM
   ISN 0014             JJ=JJ+IM
   ISN 0015             DO 100 I=1,IM
   ISN 0016             IJK=I+JJ+KK
   ISN 0017             CP1(IJK)= CP1(IJK) -DTAKA*C(IJK)
   ISN 0018         100 CONTINUE
                  C
   ISN 0019             RETURN
                  C     DEBUG SUBCHK
   ISN 0020             END

 *OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

 *OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

 *STATISTICS*     SOURCE STATEMENTS =      19 ,PROGRAM SIZE =      536

 *STATISTICS*  NO  DIAGNOSTICS GENERATED

 ****** END OF COMPILATION ******                              121K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                         OS/360  FORTRAN H

         COMPILER OPTIONS   NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002              SUBROUTINE CONSIN (IM,JM,KM,KN,IMJM,IJKM,IJKN)
                C
                C.... THIS SUBROUTINE SPECIFY VALUE OF CONSTANTS.
                C
  ISN 0003              DIMENSION CARD(20)
                C
  ISN 0004              COMMON    /AADATA/
                     *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                     *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                     *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                     *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                     *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                     *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                C
  ISN 0005              AK=0.35
  ISN 0006              A1(1)=4.7
  ISN 0007              A1(2)=15.0
  ISN 0008              A1(3)=9.0
  ISN 0009              A1(4)=0.74
  ISN 0010              ITMHR=0
  ISN 0011              ITOTHR=0
  ISN 0012              ITSTEP=0
  ISN 0013              TSEC=0.
  ISN 0014              ITSEC=TSEC
  ISN 0015              ITM=TM
  ISN 0016              IM1=IM-1
  ISN 0017              JM1=JM-1
  ISN 0018              KM1=KM-1
  ISN 0019              IMJM=IM*JM
  ISN 0020              IJKM=IM*JM*KM
  ISN 0021              IJKN=IM*JM*KN
  ISN 0022              LPRINT=3600
  ISN 0023              IF (LPRINT .GT. LTSOUS) LPRINT=LTSOUS
  ISN 0025              IF (LPRINT .GT. LTWIND) LPRINT=LTWIND
                C
  ISN 0027              RETURN
                C
                C       ---------------------------------------------------------.
                C
  ISN 0028              ENTRY CDTOTP (JUNIT1,IUNIT1,IUNIT2)
                C
                C *** PRINT CARD IMAGES OF NAMELIST ***
                C       IUNIT1=UNIT FOR CARD READER,IUNIT2=TEMPORARY DISK,JUNIT1=PRINTER..
                C
  ISN 0029              WRITE (JUNIT1, 5)
  ISN 0030            5 FORMAT ('1')
  ISN 0031           10 READ (IUNIT1, 20, END   100) (CARD(I), I   1, 20, 1)
  ISN 0032           20 FORMAT (20A4)
  ISN 0033              WRITE (IUNIT2, 20) (CARD(I), I   1, 20, 1)
  ISN 0034              WRITE ( JUNIT1, 30) (CARD(I), I = 1, 20, 1)
  ISN 0035           30 FORMAT (10X, 20A4)
  ISN 0036              GO TO 10
  ISN 0037          100 END FILE IUNIT2
  ISN 0038              REWIND IUNIT2
  ISN 0039              WRITE (JUNIT1, 200)
```

```
   ISN 0040        200 FORMAT ('0   CARDS COPIED TO DISK.')
                C
   ISN 0041            RETURN
                C      DEBUG SUBCHK
   ISN 0042            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      41 ,PROGRAM SIZE =      1076

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              117K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

          COMPILER OPTIONS   NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
 ISN 0002                SUBROUTINE DIMENS (DX,DY,DZ,RDX,RDY,RDZ,DXS,DYS,DZS,Z,ZM,
                        *   IM,JM,KM)
                  C
                  C.... THIS SUBROUTINE COMPUTES NECESSARY CONSTANT PARAMETERS FOR
                  C      NUMERICAL GRID SYSTEM EITHER CONSTANT OR VARIABLE GRID ...
                  C.... THE ROUTINE IS CALLED IN TWO WAYS. DIMENS FOR INITIALIZATION;
                  C....  DIMEN1 FOR EVERY HOUR WHEN MIXING HEIGHT IS CHANGED.
                  C
 ISN 0003                DIMENSION
                        *     DX(IM),RDX(IM),DXS(IM),DY(JM),RDY(JM),DYS(JM)
                        *    ,DZ(KM),RDZ(KM),DZS(KM),Z(KM),ZM(KM)
                  C
 ISN 0004                COMMON     /AADATA/
                        *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                        *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                        *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                        *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                        *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                        *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                  C
                  C *** INITIALIZE NUMERICAL GRID SYSTEM ***
                  C
                  C ... DXS=DX**2,RDX=RATIO OF VARIABLE GRID ,Z=ACTUAL HEIGHT
                  C     ZM=MIDDLE POINT OF GRID
                  C
                  C.... GRID IN X-DIRECTION
 ISN 0005                DO 40 I=1,IM1
 ISN 0006                DXS(I)= DX(I)*DX(I)
 ISN 0007                RDX(I+1)= DX(I+1)/DX(I)
 ISN 0008             40 CONTINUE
 ISN 0009                RDX(1)=1.0
 ISN 0010                DXS(IM)=DX(IM)*DX(IM)
                  C
                  C.... GRID IN Y-DIRECTION
 ISN 0011                DO 50 J=1,JM1
 ISN 0012                RDY(J+1)= DY(J+1)/DY(J)
 ISN 0013                DYS(J)= DY(J)*DY(J)
 ISN 0014             50 CONTINUE
 ISN 0015                RDY(1)=1.0
 ISN 0016                DYS(JM)=DY(JM)*DY(JM)
                  C
                  C.... GRID IN Z-DIRECTION
 ISN 0017                DO 60 K=1,KM1
 ISN 0018                RDZ(K+1)= DZ(K+1)/DZ(K)
 ISN 0019                DZS(K+1)= 0.5*DZ(K+1)*(DZ(K+1)+DZ(K))
 ISN 0020             60 CONTINUE
 ISN 0021                DZS(1)= DZ(1)*DZ(1)
 ISN 0022                RDZ(1)= 1.0
 ISN 0023                DZS(KM)= DZ(KM)*DZ(KM)
                  C
                  C.... GRID HEIGHT
 ISN 0024                Z(1)=0.0
 ISN 0025                ZM(1)= 0.5*DZ(1)
 ISN 0026                DO 70 K=2,KM
 ISN 0027                Z (K)= Z(K-1)+DZ(K-1)
```

```
ISN 0028              ZM(K)= 0.5*(DZ(K)+DZ(K-1))+ ZM(K-1)
ISN 0029           70 CONTINUE
ISN 0030              IF (Z(KM) .GT. HG) HG=Z(KM)
ISN 0032              HH=HG
ISN 0033              PARM(5)=HH
                C
ISN 0034              RETURN
                C
                C     ----------------------------------------------------
                C
ISN 0035              ENTRY DIMEN1 (DZ,RDZ,DZS,Z,ZM,KM,KNN,RH)
                C
                C *** CHANGE VERTICAL GRID DUE TO CHANGE IN MIXING HEIGHT ***
                C.... HH IS MIXING HEIGHT. RH IS RATIO OF NEW DZ TO OLD DZ.
                C     THE FIRST 9 GRIDS ARE FIXED,AND 4 TOP GRIDS VARY
                C     IF HH LESS THAN 300 M, Z(JM) IS KEPT AT 300M
                C     BUT THE EFFECTS OF INVERSION IS IN THROUGH DIMINISH THE EDDY K
                C
ISN 0036              HH=PARM(5)
ISN 0037              DZKM=DZ(KM)
ISN 0038              DZT=0.25*(HH-Z(KNN))
ISN 0039              IF ( DZT .LT. DZ(KNN-1)) DZT=DZ(KNN-1)
                C
ISN 0041              DO 100 K=KNN,KM
ISN 0042              DZ(K)= DZT
ISN 0043              RDZ(K)= DZ(K)/DZ(K-1)
ISN 0044              Z(K)= Z(K-1)+DZ(K-1)
ISN 0045              ZM(K)= Z(K)+0.5*DZ(K)
ISN 0046              DZS(K)= 0.5*DZ(K)*(DZ(K)+DZ(K-1))
ISN 0047          100 CONTINUE
                C
                C.... COMPUTE RATIO OF NEW &OLD VERTICAL GRID
ISN 0048              RH=DZ(KM)/DZKM
ISN 0049              IF (ITM .EQ. 0) RH=1.0
                C
ISN 0051              RETURN
                C
                C     ----------------------------------------------------------.
                C
ISN 0052              ENTRY       HHCALC      (HH,HMIN,HMAX)
                C
                C.... CALCULATE MIXING HEIGHT HH AS FUNCTION OF (IHR,HMIN,HMAX).
                C
ISN 0053              IF (IHR .LE. 5) HH=HMIN
ISN 0055              IF((IHR.GE.6).AND.(IHR.LE.13))HH=HMIN+(HMAX-HMIN)*(IHR-5.5)/8.
ISN 0057              IF((IHR.GE.14).AND.(IHR.LE.17)) HH=HMAX
ISN 0059              IF(IHR.GE.18)HH=HMAX+(HMIN-HMAX)*(IHR-17.5)/6.
                C
ISN 0061              RETURN
                C     DEBUG SUBCHK
ISN 0062              END
*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       61 ,PROGRAM SIZE =     2160

*STATISTICS*  NO  DIAGNOSTICS GENERATED
```

```
LEVEL 21.6 ( MAY 72 )                           OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE      DTTEST
                       * (U,V,AKH,AKZ,AKF,AKAHR,UZF,VZF,DX,DY,Z,IM,JM,KM,IMJM,IJKN,KN)
                  C
                  C.... THIS SUBROUTINE COMPUTES AND CHECK TIME STEP FOR NUMERICAL
                  C       STABILITY CRITERIA.
                  C
  ISN 0003              DIMENSION
                       *  U(IJKN),V(IJKN),DX(IM),DY(JM),Z(KM),AKH(KM),UZF(KM),VZF(KM)
                       *  ,AKZ(IMJM),AKF(KM)
                  C
  ISN 0004              COMMON    /AADATA/
                       *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                       *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                       *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                       *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                       *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                       *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                  C
  ISN 0005              DATA THR/3600./
                  C
  ISN 0006              LX=1
  ISN 0007              LY=IM
  ISN 0008              DTI= 1.0/THR
  ISN 0009              KKMAX=(KN-1)*IMJM
  ISN 0010              KK=-IMJM
  ISN 0011              KNMAX=KM
  ISN 0012              IF( KN .GT. 1) KNMAX=KN
  ISN 0014              DO 120 K=1,KNMAX
  ISN 0015              KK=KK+ IMJM
  ISN 0016              IF (KK .GT. KKMAX) KK=KKMAX
  ISN 0018              JJ=-IM
  ISN 0019              DO 110 J=1,JM1
  ISN 0020              JJ=JJ+ IM
  ISN 0021              DO 100 I=1,IM1
  ISN 0022              IJ= I+JJ
  ISN 0023              IJK = IJ+KK
  ISN 0024              DTU= U(IJK)*UZF(K)/DX(I)
  ISN 0025              DTV= V(IJK)*VZF(K)/DY(J)
  ISN 0026              DTU= ABS(DTU)
  ISN 0027              DTV= ABS(DTV)
  ISN 0028              DTI   = AMAX1 (DTU, DTV, DTI)
  ISN 0029          100 CONTINUE
  ISN 0030          110 CONTINUE
  ISN 0031          120 CONTINUE
  ISN 0032              DTIAKH=DTI
                  C
  ISN 0033              DYMIN=DY(1)
  ISN 0034              DXMIN=DX(1)
  ISN 0035              DO 210 I=2,IM
  ISN 0036          210 IF (DXMIN .GT. DX(I)) DXMIN=DX(I)
  ISN 0038              DO 220 J=2,JM
  ISN 0039          220 IF (DYMIN .GT. DY(J)) DYMIN=DY(J)
  ISN 0041              DXYMIN=DXMIN
  ISN 0042              IF(DXYMIN .GT. DYMIN) DXYMIN=DYMIN
                  C
```

```
ISN 0044          AKHMAX=AKH(1)
ISN 0045          AKFMAX=AKF(1)
ISN 0046          DO 300 K=2,KM
ISN 0047          IF(AKFMAX .LT. AKF(K)) AKFMAX=AKF(K)
ISN 0049          IF(AKHMAX .LT. AKH(K)) AKHMAX=AKH(K)
ISN 0051      300 CONTINUE
ISN 0052          AKHDXY=4.0*AKHMAX*AKFMAX/(DXYMIN**2)
ISN 0053          AKZMAX=AKZ(1)
ISN 0054          DO 320 I=2,IMJM
ISN 0055          IF(AKZMAX .LT. AKZ(I)) AKZMAX=AKZ(I)
ISN 0057      320 CONTINUE
            C
ISN 0058          DTIAKH=AKHDXY*AKZMAX
ISN 0059          DTIAKA=AKAHR
ISN 0060          DTI=AMAX1(DTI,DTIAKH,DTIAKA)
            C
ISN 0061          IF ( DTI .LE.  0.0) DTI= 1.0
ISN 0063          IDT= 1.0/DTI
ISN 0064          IF ( IDT .LT. 1) IDT=1
ISN 0066          IDTOT=THR
            C
            C.... CHOOSE DT SUCH THAT THR IS MULTIPLE OF DT
ISN 0067          NDT= IDTOT/IDT
ISN 0068      400 IF ((NDT*IDT) .GE. IDTOT)  GO TO 420
ISN 0070      410 IDT= IDT-1
ISN 0071          NDT= IDTOT/IDT
ISN 0072          GO TO 400
ISN 0073      420 CONTINUE
ISN 0074          DT=IDT
ISN 0075          ITT=TM/IDTOT
ISN 0076          TI=TM-ITT*IDTOT
ISN 0077          DTLAST=IDTOT-TI
ISN 0078          IF (DT .GT. DTLAST) DT=DTLAST
            C
ISN 0080          RETURN
            C     DEBUG SUBCHK
ISN 0081          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       80 ,PROGRAM SIZE =      1866

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                               109K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                   C
  ISN 0002               SUBROUTINE       GEOIN
                        &(QB,ZS,Z0,PQB,XPUTM,YPUTM,XP,YP,ZP,ZR,NP,IUTM,JUTM,DX,DY,IM,JM
                        & ,LM,LMAX,XSUTM,YSUTM,XS,YS,IS,JXS,JYS,NS,NRAMS,KUNITG,JUNIT
                        & ,QBSUM,PQBSUM,QBTOT,PQBTOT,ZOMEAN)
                   C
                   C.... THIS ROUTINE READ IN GEOGRAPHICAL AND ANNUAL EMISSION DATA.
                   C     IT ALSO: 1) FIX UTM COORDINATES FOR ALL NUMERICAL GRIDS.
                   C              2) CONVERT LOCATION OF RAMS STATION TO NUMERICAL GRID.
                   C              3) CONVERT LOCATION OF POINT SOURCE TO NUMERICAL GRID.
                   C
  ISN 0003               DIMENSION
                        *  QB(IM,JM),ZS(IM,JM),Z0(IM,JM)
                        *  ,PQB(LM),XPUTM(LM),YPUTM(LM),XP(LM),YP(LM),ZP(LM),ZR(LM),NP(LM)
                        *  ,IS(NS),XSUTM(NS),YSUTM(NS),JXS(NS),JYS(NS),XS(NS),YS(NS)
                        *  ,IUTM(IM),JUTM(JM),DX(IM),DY(JM)
  ISN 0004               DIMENSION FMTDS(10)
                   C
  ISN 0005               DATA DXA/1000.0/,DYA/1000.0/
                   C
  ISN 0006          8000 FORMAT (4I5)
  ISN 0007          8010 FORMAT (10A4)
  ISN 0008          8050 FORMAT (F10.4)
  ISN 0009          8090 FORMAT (F10.1)
                   C
                   C.... READ ORGIN OF UMT COORDINATE AND FIX FOR ALL GRIDS.
  ISN 0010               READ (KUNITG,8000) IXBEG,IYBEG,IBEG,JBEG
                   C
  ISN 0011               CALL      XYUTMS
                        & (IUTM,JUTM,IXBEG,IYBEG,IBEG,JBEG,DX,DY,DXA,DYA,IM,JM)
                   C
                   C
                   C.... READ IN UTM COORDINATES OF RAMS STATIONS.
  ISN 0012               READ (KUNITG,8000) NRAMS
  ISN 0013               READ (KUNITG,8010) FMTDS
  ISN 0014               READ (KUNITG,FMTDS) (IS(L),XSUTM(L),YSUTM(L),L=1,NRAMS)
                   C
                   C.... CONVERT STATION FROM UTM UNIT (XSUTM,YSUTM) TO GRID UNIT(XS,YS).
                   C
  ISN 0015               CALL      XYUTM1      (XSUTM,YSUTM,NS,XS,YS,NS,1)
                   C
  ISN 0016               DO 100 L=1,NRAMS
  ISN 0017               JXSI=XS(L)+1.0001
  ISN 0018               JYSI=YS(L)+1.0001
  ISN 0019               IF (JXSI .LT. 1) JXSI=1
  ISN 0021               IF (JXSI .GT. IM) JXSI=IM
  ISN 0023               IF (JYSI .LT. 1) JYSI=1
  ISN 0025               IF (JYSI .GT. JM) JYSI=JM
  ISN 0027               JXS(L)=JXSI
  ISN 0028               JYS(L)=JYSI
  ISN 0029           100 CONTINUE
                   C
                   C.... READ IN SURFACE ROUGHNESS
  ISN 0030               READ (KUNITG,8010) FMTDS
  ISN 0031               READ (KUNITG,FMTDS) Z0
  ISN 0032               IF (ZOMEAN.LE.0.0) GO TO 120
```

```
ISN 0034          DO 110 J=1,JM
ISN 0035          DO 110 I=1,IM
ISN 0036           ZO(I,J)=ZOMEAN
ISN 0037      110 CONTINUE
ISN 0038          DO 115 J=11,27
ISN 0039          DO 115 I=5,14
ISN 0040          ZO(I,J)= 2.0
ISN 0041      115 CONTINUE
ISN 0042      120 CONTINUE
            C
            C.... READ IN AREA SOURCE EMISSION HEIGHT
ISN 0043          READ (KUNITG,8010) FMTDS
ISN 0044          READ (KUNITG,FMTDS) ZS
            C
            C.... READ IN POINT SOURCES LOCATION AND STACK HEIGHT.
ISN 0045          READ (KUNITG,8000) LMAX
ISN 0046          READ (KUNITG,8010) FMTDS
ISN 0047          READ (KUNITG,FMTDS) (NP(L),XPUTM(L),YPUTM(L),ZP(L),L=1,LMAX)
            C
            C.... READ IN AREA SOURCE EMISSION RATES
ISN 0048          READ (KUNITG,8010) FMTDS
ISN 0049          READ (KUNITG,FMTDS) QB
ISN 0050          READ (KUNITG,8050) QBTOT
            C
            C.... OBTAIN AVERAGE EMISSION RATE FOR WINTER MONTHS
ISN 0051          DO 150 J=1,JM
ISN 0052          DO 150 I=1,IM
ISN 0053      150 QB(I,J)=QB(I,J)*4.0
ISN 0054          QBSUM=QBTOT*4.0
            C
            C.... READ IN POINT SOURCE EMISSION RATES.
ISN 0055          READ (KUNITG,8010) FMTDS
ISN 0056          READ (KUNITG,FMTDS) (PQB(L),L=1,LMAX)
ISN 0057          READ (KUNITG,8090) PQBTOT
            C
            C.... READ IN PLUME RISE DATA.
ISN 0058          READ (KUNITG,8010) FMTDS
ISN 0059          READ (KUNITG,FMTDS) (ZR(L),L=1,LMAX)
            C
            C.... CONVERT XP,YP FROM UTM COORDINATE TO NUMERICAL GRID
ISN 0060          PQBSUM=0.0
ISN 0061          DO 200 L=1,LMAX
ISN 0062          IF (NP(L) .EQ. 39) PQB(L)=0.0
ISN 0064          PQBSUM=PQBSUM+PQB(L)
ISN 0065      200 CONTINUE
            C
            C.... CONVERT POINT SOURCE FROM UTM UNIT(XPUTM,YPUTM) TO GRID UNIT(XP,YP
            C
ISN 0066          CALL      XYUTM1      (XPUTM,YPUTM,LM,XP,YP,LM,1)
            C
ISN 0067          RETURN
            C     DEBUG SUBCHK
ISN 0068          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      67 ,PROGRAM SIZE =     2934
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE OUTAPE (CC,ICAL,IOBS,NSX,IM,JM,IDAYTP,IHRTP,LWARM)
                C
                C     THIS SUBROUTINE WRITES COMPUTED RESULTS ON TWO I/O UNITS
                C      UNIT=KUNITP- PARM,CC,ICAL,IOBS.
                C      UNIT=KUNITC- PARM,ICAL,IOBS.
                C
  ISN 0003            DIMENSION CC(IM,JM),ICAL(NSX),IOBS(NSX)
                C
  ISN 0004            COMMON    /AADATA/
                    *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                    *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                    *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                    *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                    *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                    *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                C
  ISN 0005        800 FORMAT (1X,4I2,F6.1,F6.0,F6.1,F6.2,F6.1,2F6.0
                     &      ,2F6.1,F9.0,4(/7X,13I5))
  ISN 0006        830 FORMAT ( '  TAPE,IMO,IDAY,IHR =   ',3I3)
  ISN 0007        831 FORMAT ('  *** ERR=291 ***')
  ISN 0008        832 FORMAT ('  *** END=292 ***')
                C
  ISN 0009            IHRP=IHR
  ISN 0010            IDAYP=IDAY
  ISN 0011            IF (IHR.EQ.0) IDAYP=IDAY-1
  ISN 0013            IF (IHR.EQ.0) IHRP=24
  ISN 0015            IF (ITM.NE.0) GO TO 310
                C
                C.... INITIALIZE OUTPUT DATA SET. (ONLY WHEN ITM=0)
  ISN 0017            LWARM=1
  ISN 0018            IF ((IDAY.EQ.IDAYTP).AND.(IHR.EQ.(IHRTP-1))) LWARM=0
                C
                C.... IF COLD START, RETURN.....
                C.... IF WARM START, GET DATA SET KUNITC AND KUNITP READY.
                C
  ISN 0020            IF (LWARM.EQ.0) RETURN
  ISN 0022            IF (LWRITE(9) .EQ. 0) GO TO 200
  ISN 0024            REWIND KUNITC
  ISN 0025        100 READ (KUNITC,800,ERR=291,END=200) KYR,KMO,KDAY,KHR,PARM,ICAL,IOBS
  ISN 0026            IF ((KDAY.NE.IDAYP) .OR. (KHR.NE.IHRP)) GO TO 100
                C.... THIS BACKSPACE IS TO PREVENT THE LOST OF POINTER IN BUFFER.
                C     DO 110 L=1,5
                C     BACKSPACE KUNITC
                C 110 CONTINUE
                C     WRITE (KUNITC,800) KYR,KMO,KDAY,KHR,PARM,ICAL,IOBS
  ISN 0028        200 CONTINUE
                C
  ISN 0029            IF (LWRITE(10) .EQ. 0) GO TO 300
  ISN 0031            REWIND KUNITP
  ISN 0032        220 READ (KUNITP,ERR=291,END=292) KYR,KMO,KDAY,KHR,PARM,CC,ICAL,IOBS
  ISN 0033            WRITE (JUNIT,830) KMO,KDAY,KHR
  ISN 0034            IF ((KDAY.NE.IDAYP).OR.(KHR.NE.IHRP)) GO TO 220
                C.... SAME REASON AS FOR BACKSPACE KUNITC
                C     BACKSPACE KUNITP
                C     WRITE(KUNITP,ERR=291,END=292)KYR,KMO,KDAY,KHR,PARM,CC,ICAL,IOBS
```

```
      ISN 0036           GO TO 300
      ISN 0037       291 WRITE (JUNIT,831)
      ISN 0038           STOP 291
      ISN 0039       292 WRITE (JUNIT,832)
      ISN 0040           STOP 292
      ISN 0041       300 CONTINUE
                   C
      ISN 0042           RETURN
                   C
                   C.... WRITE STATMENTS.
                   C
      ISN 0043       310 CONTINUE
                   C
      ISN 0044           IF (LWRITE(9).EQ.0) GO TO 320
      ISN 0046           WRITE (KUNITC,800) IYR,IMO,IDAYP,IHRP,PARM,ICAL,IOBS
      ISN 0047       320 CONTINUE
      ISN 0048           IF (LWRITE(10).EQ.0) GO TO 340
      ISN 0050           WRITE (KUNITP) IYR,IMO,IDAYP,IHRP,PARM,CC,ICAL,IOBS
      ISN 0051       340 CONTINUE
                   C
                   C *** INITIALIZE CC ARRAY ***
      ISN 0052           DO 400 J=1,JM
      ISN 0053           DO 400 I=1,IM
      ISN 0054       400 CC(I,J)=0.
                   C
      ISN 0055           RETURN
                   C     DEBUG SUBCHK
      ISN 0056           END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      55 ,PROGRAM SIZE =      1806

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              117K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                 C
   ISN 0002            SUBROUTINE POSITV (   A, IM, JM, KM, IMJM, IJKM)
                 C
                 C.... THIS SUBROUTINE  SET A(IJK)=0., IF IT HAS A NEGATIVE VALUE.
                 C
   ISN 0003            DIMENSION A(IJKM)
                 C
   ISN 0004            KK=-IMJM
   ISN 0005            DO 10 K=1,KM
   ISN 0006            KK=KK+IMJM
   ISN 0007            JJ=-IM
   ISN 0008            DO 10 J=1,JM
   ISN 0009            JJ=JJ+IM
   ISN 0010            DO 10 I=1,IM
   ISN 0011            IJK=I+JJ+KK
   ISN 0012            IF ( A(IJK) .LT. 0.0) A(IJK)=0.0
   ISN 0014         10 CONTINUE
                 C
   ISN 0015            RETURN
                 C     DEBUG SUBCHK
   ISN 0016            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS       15 ,PROGRAM SIZE =       408

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                         125K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                          OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                 C
  ISN 0002             SUBROUTINE PRINTS (CP1,CC,C1,COLD,IOBS,ICAL,KOBS,KCAL,ITOBS
                      *  ,U,V,W,U1,V1,UU,VV,AKZ,UZF,VZF,WZF,AKF,DX,DY
                      *  ,IUTM,JUTM,XSUTM,YSUTM,XPUTM,YPUTM,ZO,ZS,QB,PQB
                      *  ,XP,YP,ZP,ZR,NP,IN,JN,KN,LM
                      *  ,IS,XS,YS,JXS,JYS,Z,IM,JM,KM,NS,NSX)
                 C
                 C.... THIS SUBROUTINE PRINTS INPUT AND OUTPUT DATA OF THE MODEL.
                 C       ENTRY PRINTA -- HOURLY SOURCE EMISSION RATES.
                 C       ENTRY PRINTB -- HOURLY METEOROLOGICAL PARAMETERS.
                 C       ENTRY PRINTC -- HOURLY CONCENTRATION VALUES.
                 C
                 C.... PRINT OPTION IS CONTROL BY LWRITE(N). (SEE TABLE IN MAIN PROGRAM)
                 C
  ISN 0003             DIMENSION
                      *       CP1(IM,JM,KM),CC(IM,JM),C1(IM,JM),COLD(IM,JM)
                      *      ,IOBS(NSX),ICAL(NSX),KCAL(NSX),KOBS(NSX),ITOBS(NSX)
                      *      ,U(IM,JM,KN),V(IM,JM,KN),W(IM,JM,KN),AKZ(IM,JM)
                      *      ,U1(IM,JM),V1(IM,JM),UU(IN,JN),VV(IN,JN)
                      *      ,UZF(KM),VZF(KM),WZF(KM),AKF(KM),Z(KM),DX(IM),DY(JM)
                      *      ,XS(NS),YS(NS),IS(NS),JXS(NS),JYS(NS),XSUTM(NS),YSUTM(NS)
                      *      ,QB(IM,JM),PQB(LM),ZS(IM,JM),ZO(IM,JM),IUTM(IM),JUTM(JM)
                      *      ,XP(LM),YP(LM),ZP(LM),ZR(LM),NP(LM),XPUTM(LM),YPUTM(LM)
                 C
  ISN 0004             COMMON    /AADATA/
                      *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                      *  ,IYR,IMO,IDAY,IHR,ITM,ITMMR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                      *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                      *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                      *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                      *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                 C
  ISN 0005             REAL*4 TITLE1(10)/2*'     ',' FIE','LD  ','UNIT',' IN '
                     &                 ,2*'     ',', RA','TIO='/
                     &          ,TITLE2(10)/2*'     ',' FIE','LD F','OR L','AYER'
                     &                 ,' K= ','     ',', RA','TIO='/
                     &          ,LAYER(14)/'  1 ','  2 ','  3 ','  4 ','  5 ','  6 ','  7 '
                     &                 ,'  8 ','  9 ',' 10 ',' 11 ',' 12 ',' 13 ',' 14 '/
                     &          ,TITLQB/'* QB'/,  TITLZO/'* ZO'/,  TITLZS/'* ZS'/
                     &          ,TITLKZ/'* KZ'/,  TITLC1/'* C1'/,  TITLCC/'* CC'/
                     &          ,TITLL1/'1 **'/,  TITLL2/'0 **'/,  TITLL3/'  **'/
                     &          ,TUNIT1/'G/S '/,  TUNIT2/' M. '/
                 C
  ISN 0006        8110 FORMAT ('1 ***** LOCATION OF ',I2,' RAMS STATIONS *****'//
                     & '    IS',4X,'XSUTM',5X,'YSUTM',6X,'XS',6X,'YS',3X,'JXS',3X,'JYS')
  ISN 0007        8120 FORMAT (I5,2F10.2,2F8.2,2I6)
  ISN 0008        8140 FORMAT (//5X,'.... TOTAL EMISSION : READ IN=',F10.1
                     &         ,',     MODEL USED=',F10.1,',   IN G/SEC')
  ISN 0009        8150 FORMAT ('1 *** POINT SOURCE DATA ***'/
                     &         5X,'..... TOTAL NUMBER =',I5,//'    N '
                     &         ,'    ID',5X,'XPUTM',5X,'YPUTM',8X,'ZP',8X,'QP'
                     &         ,8X,'ZR',6X,'XP',6X,'YP')
  ISN 0010        8160 FORMAT (2I5,3F10.1,F10.2,F10.1,2F8.2)
                 C
  ISN 0011             QBSUM=PARM(6)
  ISN 0012             PQBSUM=PARM(7)
```

```
                C
                C *** PRINT GEOGRAPHICAL AND ANNUAL EMISSION DATA ***
                C
                C.... INITIALIZE SUBROUTINE WRITES, WHICH HAS ENTRY WRITEX.
                C
ISN 0013              CALL       WRITES
                     &       (QB,IM,JM,IM,JM,1,1,IUTM,JUTM,IM,JUNIT,RATIO,TITLE1)
                C
ISN 0014              IF (LWRITE(1).EQ.0) RETURN
                C
                C.... RAMS STATIONS.
                C
ISN 0016              WRITE (JUNIT,8110) NS
ISN 0017              WRITE (JUNIT,8120)
                     &  (IS(L),XSUTM(L),YSUTM(L),XS(L),YS(L),JXS(L),JYS(L),L=1,NS)
                C
                C.... SURFACE ROUGHNESS.
ISN 0018              RATIO=100.
ISN 0019              TITLE1(1)=TITLL1
ISN 0020              TITLE1(2)=TITLZO
ISN 0021              TITLE1(7)=TUNIT2
                C
ISN 0022              CALL       WRITEX
                     &       (ZO,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE1)
                C
                C.... EMMISION HEIGHT OF AREA SOURCE.
                C
ISN 0023              RATIO=1.0
ISN 0024              TITLE1(2)=TITLZS
                C
ISN 0025              CALL        WRITEX
                     &       (ZS,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE1)
                C
                C.... EMISSION RATE OF AREA SOURCE.
                C
ISN 0026              RATIO=1.0
ISN 0027              TITLE1(2)=TITLQB
ISN 0028              TITLE1(7)=TUNIT1
                C
ISN 0029              CALL       WRITEX
                     &       (QB,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE1)
                C
ISN 0030              WRITE (JUNIT,8140) QBTOT,QBSUM
                C
                C.... POINT SOURCE DATA
                C
ISN 0031              WRITE (JUNIT,8150) LM
ISN 0032              WRITE (JUNIT,8160)
                     &(L,NP(L),XPUTM(L),YPUTM(L),ZP(L),PQB(L),ZR(L),XP(L),YP(L),L=1,LM)
ISN 0033              WRITE (JUNIT,8140) PQBTOT,PQBSUM
                C
ISN 0034              RETURN
                C
                C     --------------------------------------------------------
                C
                C *** PRINTS TIME VARYING SOURCE EMISSION RATES ***
                C
ISN 0035              ENTRY PRINTA
```

```
              C
ISN 0036       8200 FORMAT (1X,'** TOTAL SOURCE QBTOT=',F10.1,' PQBTOT=',F10.1
                  &       ,' QBSUM=',F10.1,' PQBSUM=',F01.1)
ISN 0037       8240 FORMAT ( /1X,'** POINT SOURCE **'/,(5(1X,I3,2F9.1)))
              C
ISN 0038            IF (LWRITE(2).LE.0) GO TO 200
ISN 0040            QBSUM=PARM(6)
ISN 0041            PQBSUM=PARM(7)
ISN 0042            WRITE (JUNIT,8200) QBTOT,PQBTOT,QBSUM,PQBSUM
ISN 0043            IF (LWRITE(2).LT.2) GO TO 200
ISN 0045            TITLE1(1)=TITLL2
ISN 0046            TITLE1(2)=TITLQB
ISN 0047            TITLE1(7)=TUNIT1
              C
ISN 0048            IF ((LWRITE(2).EQ.2).OR.(LWRITE(2).EQ.4))     CALL     WRITEX
                  & (QB,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE1)
              C
ISN 0050            IF (LWRITE(2).GE.3) WRITE (JUNIT,8240) (L,PQB(L),ZR(L),L=1,LM)
ISN 0052        200 CONTINUE
              C
ISN 0053            RETURN
              C
              C     ---------------------------------------------------
              C
              C *** PRINTS METEOROLOGICAL PARAMETERS AND MODEL DATA ***
              C
ISN 0054            ENTRY PRINTB (RH)
              C
ISN 0055       8300 FORMAT (1X,'    ***** RAMS DATA *****'/'  IS ',25I5/
                  &      '  UU ',25F5.1/'  DD ',25F5.0/'  T1 ',25F5.0/
                  &      '  T2 ',25F5.0/'  CC ',25F5.0/'  RA ',25F5.0)
ISN 0056       8310 FORMAT (1X,'    ***** ANALYZED WIND UU(IN,JN) & VV(IN,JN) *****'/
                  &        ,5X,'J',9(' ----',I2,'---- '))
ISN 0057       8320 FORMAT (1X,I5,18F6.1)
ISN 0058       8400 FORMAT (1X,' ** U,V WIND COMPONENTS FOR LAYER K=',I2,' **'/
                  &        ,5X,'J',10(' ----',I2,'---- '))
ISN 0059       8410 FORMAT (1X,I5,20F6.1)
ISN 0060       8420 FORMAT (1X,' ** W WIND COMPONENT FOR LAYER K=',I2,' **'/
                  &        ,5X,'J',10(' ----',I2,'---- '))
ISN 0061       8430 FORMAT (1X,I5,10(1X,F10.4,1X))
ISN 0062       8450 FORMAT (/,' RH=',F6.2,'  GRID Z(K)= ',14F6.0)
ISN 0063       8460 FORMAT (15X,'UZF(K)= ',14F6.2)
ISN 0064       8470 FORMAT (15X,'VZF(K)= ',14F6.2)
ISN 0065       8480 FORMAT (15X,'WZF(K)= ',14F6.2)
ISN 0066       8500 FORMAT (15X,'AKF(K)= ',14F6.2)
ISN 0067       8510 FORMAT (11X,'KZ(J=JM/2)= ',15F6.1)
ISN 0068       8600 FORMAT (' DT=',F6.1,4X,'PARM(N)= ',10F10.2)
ISN 0069       8610 FORMAT (3X,'U0=',F8.3,'  PHZ=',F8.3,'  HFZ=',F8.3,'  RIB=',F8.3)
              C
              C.... RAMS DATA.
ISN 0070            IF (LWRITE(3).GE.1)
                  &      WRITE (JUNIT,8300) IS,((RAMS(I,J),J=1,NS),I=1,6)
              C
              C.... ANALYZED WIND FIELD.
ISN 0072            IF ((KWIND.NE.2).OR.(KWIND.NE.4)) GO TO 320
ISN 0074            IF (LWRITE(3).NE.2) GO TO 320
ISN 0076            WRITE (JUNIT,8310) (I,I=1,IN)
ISN 0077            DO 310 JJ=1,JN
```

```
ISN 0078          J=JN+1-JJ
ISN 0079          WRITE (JUNIT,8320) J,(UU(I,J),VV(I,J),I=1,IN)
ISN 0080      310 CONTINUE
ISN 0081      320 CONTINUE
            C
            C.... WIND FIELD AT NUMERICAL GRIDS.
ISN 0082          IF (LWRITE(4).EQ.0) GO TO 500
ISN 0084          IF (LWRITE(4).LT.2) GO TO 440
ISN 0086          K=1
ISN 0087          WRITE (JUNIT,8400) K,(I,I=2,IM,3)
ISN 0088          DO 410 JJ=1,JM,4
ISN 0089          J= JM+1-JJ
ISN 0090      410 WRITE (JUNIT,8410) J,(U(I,J,1),V(I,J,1),I=2,IM,3)
ISN 0091          IF (LWW.NE.1) GO TO 420
ISN 0093          WRITE (JUNIT,8420) K,(I,I=2,IM,3)
ISN 0094          DO 415 JJ=1,JM,4
ISN 0095          J=JM+1-JJ
ISN 0096      415 WRITE (JUNIT,8430) J,(W(I,J,1),I=2,IM,3)
ISN 0097      420 CONTINUE
ISN 0098          IF (LWRITE(4).NE.3) GO TO 440
ISN 0100          DO 430 K=2,KN
ISN 0101          WRITE (JUNIT,8400) K,(I,I=2,IM,3)
ISN 0102          DO 430 JJ=1,JM,4
ISN 0103           J=JM+1-JJ
ISN 0104          WRITE (JUNIT,8410) J,(U(I,J,K),V(I,J,K),I=2,IM,3)
ISN 0105      430 CONTINUE
ISN 0106           IF (LWW.NE.1) GO TO 440
ISN 0108          DO 435 K=2,KN
ISN 0109          WRITE (JUNIT,8420)  K,(I,I=2,IM,3)
ISN 0110          DO 435 JJ=1,JM,4
ISN 0111          J=JM+1-JJ
ISN 0112          WRITE (JUNIT,8430) J,(W(I,J,K),I=2,IM,3)
ISN 0113      435 CONTINUE
ISN 0114      440 CONTINUE
            C
            C.... VERTICAL GRIDS AND WIND PROFILE
ISN 0115          WRITE (JUNIT,8450) RH,(Z(K),K=1,KM)
ISN 0116          WRITE (JUNIT,8460) (UZF(K),K=1,KM)
ISN 0117          WRITE (JUNIT,8470) (VZF(K),K=1,KM)
ISN 0118          IF (LWW.EQ.1) WRITE (JUNIT,8430) (WZF(K),K=1,KM)
ISN 0120      500 CONTINUE
            C
            C.... EDDY DIFFUSIVITY.
ISN 0121          IF (LWRITE(5).GE.1) WRITE (JUNIT,8500) (AKF(K),K=1,KM)
ISN 0123          JJ=JM/2
ISN 0124          IF (LWRITE(5).EQ.2)
                 &   WRITE (JUNIT,8510) (AKZ(I,JJ),I=1,IM,2)
ISN 0126          RATIO=10.0
ISN 0127          TITLE2(1)=TITLL2
ISN 0128          TITLE2(2)=TITLKZ
ISN 0129          TITLE2(8)=LAYER(1)
            C
ISN 0130          IF (LWRITE(5).EQ.3)
                 &   CALL WRITEX (AKZ,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE2)
            C
            C.... METEOROLOGICAL PARAMETERS.
ISN 0132          IF (LWRITE(6).EQ.0) GO TO 600
ISN 0134          WRITE (JUNIT,8600) DT,PARM
```

```
ISN 0135          IF (LWRITE(6).EQ.2) WRITE (JUNIT,8610) UO,PHIFHZ,HFZ,RIB
ISN 0137      600 CONTINUE
            C
ISN 0138          RETURN
            C
            C     ------------------------------------------------------
            C
            C *** PRINT CONCENTRATION FIELD ***
            C
ISN 0139          ENTRY PRINTC (IMC,JMC   )
            C
            C.... THIS ROUTINE PRINTS (ACCORDING TO CONTROL FLAG) :
            C       1) INSTANTANEOUS CP1 AT STATIONS FOR EACH TIME STEP;
            C       2) INSTANTANEOUS CP1 FIELD AT EACH TIME INTERVAL 'LPRINT';
            C       3) AVERAGE CONC. CC FIELD AT EACH TIME INTERVAL 'LPRINT';
            C       4) AVERAGE CONC. CC AT STATIONS (IOBS,ICAL); AND
            C       5) 24 HOURS AVERAGE CONC. AT STATIONS (KOBS,KCAL).
            C     NS = NO. OF RAMS STATION.  NSX = NS+1 .
            C     NSX IS INDEX FOR SPATIAL AVERAGE. F(NSX)= AVERAGE OF F(L),L=1,NS.
            C
ISN 0140          DATA ITPR/0/
            C
ISN 0141     8700 FORMAT (/3X,'TIME',3X,' SEC NTS',4X,'DT',2X,25I4)
ISN 0142     8710 FORMAT (1X,I2,2('/',I2),I5,I4,F6.0,2X,25I4)
ISN 0143     8800 FORMAT (//,' ** 2HR STATION SO2 ** MO,DAY,HR=',3(I2,'/'),',  NTS='
                &       ,I4,',    ... SPATIAL AVERAGE... CAL=',I5,' OBS=',I5)
ISN 0144     8810 FORMAT (//,' ** 24HR STATION SO2 ** MO,DAY,HR=',3(I2,'/')
                &       ,',   ... SPATIAL AVERAGE... CAL=',I5,' OBS=',I5)
ISN 0145     8820 FORMAT ('  IS ',25I5/'  XS ',25F5.1/'. YS ',25F5.1/
                &       ,' CAL ',25I5,/' OBS ',25I5)
            C
            C *** PRINT INSTANTANEOUS CONCENTRATION AT MONITORING STATIONS ***
            C.... THESE VALUES ARE FROM NEAREST SOUTHWEST GRID TO STATIONS.
ISN 0146          RATIO=1.0
ISN 0147          IF ((LWRITE(7).NE.2).AND.(LWRITE(7).NE.4)) GO TO 730
ISN 0149          IF (ITPR.GT.0) GO TO 710
ISN 0151          ITPR=1
ISN 0152          WRITE (JUNIT,8700) (IS(L),L=1,NS)
ISN 0153      710 CONTINUE
ISN 0154          DO 720 L=1,NS
ISN 0155          I=JXS(L)
ISN 0156          J=JYS(L)
ISN 0157      720 ICAL(L)=CP1(I,J,1)
ISN 0158          WRITE (JUNIT,8710) IMO,IDAY,IHR,ITSEC,ITSTEP,DT,(ICAL(L),L=1,NS)
ISN 0159      730 CONTINUE
            C
ISN 0160          IF (LCRUN .EQ. 0) RETURN
            C
            C *** PRINT INSTANTANEOUS CONC. CP1 FIELD EVERY LPRINT INTERVAL ***
            C
            C.... IF NOT ON THE HOUR, RETURN TO TIME STEP LOOP OF MAIN PROGRAM.
ISN 0162          IF (ITM .NE. (ITM/LPRINT*LPRINT)) RETURN
            C
ISN 0164          ITPR=0
ISN 0165          IF (LWRITE(7).LE.0) GO TO 770
ISN 0167          IF (LWRITE(7).LE.2) GO TO 760
ISN 0169          KB=1
ISN 0170          KA=1
```

```
ISN 0171          IF (IHR.EQ.0) KB=KM
            C
            C.... PRINT HORIZONTAL FIELD OF CP1.
ISN 0173          TITLE2(1)=TITLL2
ISN 0174          TITLE2(2)=TITLC1
ISN 0175          DO 750 K=KA,KB
ISN 0176          TITLE2(8)=LAYER(K)
ISN 0177          DO 740 J=1,JM
ISN 0178          DO 740 I=1,IM
ISN 0179          COLD(I,J)=CP1(I,J,K)
ISN 0180      740 CONTINUE
            C
ISN 0181          CALL      WRITEX     (COLD,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE2)
            C
ISN 0182      750 CONTINUE
            C
            C.... PRINT VERTICAL CROSS SECTION OF CP1 AT I=IMC, J=JMC.
ISN 0183      760 CONTINUE
            C
ISN 0184          CALL      WRITEZ     (CP1,Z,RATIO,IM,JM,KM,IMC,JMC,TITLC1)
            C
ISN 0185      770 CONTINUE
            C
            C *** PRINT CC FIELD EVERY LPRINT INTERVAL ***
            C.... CC IS AVERAGE SURFACE CONCENTRATION FIELD OF LPRINT INTERVAL.
ISN 0186          TITLE2(1)=TITLL1
ISN 0187          TITLE2(2)=TITLCC
ISN 0188          TITLE2(8)=LAYER(1)
            C
ISN 0189          IF (LWRITE(8).GE.2)     CALL      WRITEX
                & (CC,IM,JM,IM,JM,IM,JUNIT,RATIO,TITLE2)
            C
            C *** OBTAIN VALUES OF CC AT RAMS STATION ***
            C
ISN 0191          CALL      STNCON
                & (CC,IOBS,ICAL,KOBS,KCAL,ITOBS,NS,NSX,IM,JM,XS,YS,JXS,JYS)
            C
            C *** PRINT VALUES OF CC AT RAMS STATION ***
ISN 0192          IF (LWRITE(8).LT. 1) GO TO 780
ISN 0194          WRITE (JUNIT,8800)  IMO,IDAY,IHR,ITSTEP,ICAL(NSX),IOBS(NSX)
ISN 0195          WRITE (JUNIT,8820) IS,XS,YS,(ICAL(L),L=1,NS),(IOBS(L),L=1,NS)
ISN 0196      780 CONTINUE
            C
            C *** PRINT 24 HR AVERAGE CONCENTRATION AT RAMS STATIONS ***
ISN 0197          IF (IHR .NE. 0 ) GO TO 910
ISN 0199          WRITE (JUNIT,8810) IMO,IDAY,IHR,KCAL(NSX),KOBS(NSX)
ISN 0200          WRITE (JUNIT,8820) IS,XS,YS,(KCAL(L),L=1,NS),(KOBS(L),L=1,NS)
            C
            C *** INITIALIZE ARRAY OF 24 HOUR AVERAGE ***
ISN 0201          DO 900 L=1,NS
ISN 0202          KOBS(L)=0
ISN 0203          KCAL(L)=0
ISN 0204      900 ITOBS(L)=0
            C
ISN 0205      910 CONTINUE
ISN 0206          RETURN
            C     DEBUG SUBCHK
ISN 0207          END
```

```
                                              FORTRAN H ERROR MESSAGES
                   ERROR NO    LEVEL        ERROR MESSAGE

    NAME   TITLL3    IEK307I        4        THE DATA STATEMENT CONTAINS A VARIABLE THAT IS NOT REFERENCED.

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*      SOURCE STATEMENTS =      206 ,PROGRAM SIZE =      9356

*STATISTICS*     1 DIAGNOSTICS GENERATED, HIGHEST SEVERITY CODE IS  4

****** END OF COMPILATION ******                                   53K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                         OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002             SUBROUTINE SHIFTN ( A, B, IM, JM, KM, IN, JN, KN)
                C
                C.... SHIFT A TO B
                C
  ISN 0003             DIMENSION A(IM,JM,KM),  B(IN,JN,KN)
                C
  ISN 0004             IMM=IN
  ISN 0005             JMM=JN
  ISN 0006             KMM=KN
  ISN 0007             IF ( KM .LE. KN) KMM=KM
  ISN 0009             IF( IM .LE. IN ) IMM=IM
  ISN 0011             IF( JM .LE. JN ) JMM=JM
  ISN 0013             DO 100 K=1,KMM
  ISN 0014             DO 100 J=1,JMM
  ISN 0015             DO 100 I=1,IMM
  ISN 0016             B (I,J,K)= A( I,J,K)
  ISN 0017         100 CONTINUE
                C
  ISN 0018             RETURN
                C      DEBUG SUBCHK
  ISN 0019             END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      18 ,PROGRAM SIZE =      810

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                          125K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

      COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                         SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
              C
 ISN 0002           SUBROUTINE      SOURCE
                   * (CP1, C,QA,QB,ZS,PQA,PQB,XPUTM,YPUTM,XP,YP,ZP,ZR,ZPR,U,V,UZF,VZF
                   *   ,AKF,AKH,AKZ,DX,DY,DZ,Z,EK,FK,IM,JM,KM,KN,IMJM,LM)
              C
              C     THIS SUBROUTINE ADDS NEW SOURCE EMISSION INTO THE SYSTEM.
              C
 ISN 0003           DIMENSION
                   *        PQA(LM),PQB(LM),QA(IM,JM),QB(IM,JM)
                   *        ,XPUTM(LM),YPUTM(LM),XP(LM),YP(LM),ZP(LM),ZR(LM),ZPR(LM)
                   *        ,CP1(IM,JM,KM), C(IM,JM,KM), ZS(IM,JM)
                   *        ,U(IM,JM,KN), V(IM,JM,KN), UZF(KM), VZF(KM)
                   *        ,DX(IM),DY(JM),DZ(KM), Z(KM),EK(LM),FK(LM)
                   *        ,AKF(KM),AKH(KM),AKZ(IM,JM)
              C
 ISN 0004           COMMON     /AADATA/
                   *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                   *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                   *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                   *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                   *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                   *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
              C
 ISN 0005           DATA QMUGM/1000000./,CT1/0.027144/
              C
              C     ---------------- AREA SOURCE ----------------
              C
 ISN 0006           IF (LSOUS(1) .EQ. 0) GO TO 200
              C
              C *** CONVERT ZS FROM (M) TO GRID UNIT ***
              C
 ISN 0008           IF (ITM.EQ.0)      CALL      ZZGRID     (ZS,ZS,Z,IMJM,IMJM,KM)
              C
              C *** ADD AREA SOURCE INTO GRID CELL ***
 ISN 0010           DO 100 J=1,JM
 ISN 0011           DO 100 I=1,IM
 ISN 0012           ZSIJ=ZS(I,J)+1.001
 ISN 0013           KZS=ZSIJ
 ISN 0014           IF (KZS .GE. KM) GO TO 100
 ISN 0016           DZSK=ZSIJ-KZS
 ISN 0017           DZK=0.5*DZ(KZS)
 ISN 0018           IF (KZS.GE. 2) DZK=0.5*(DZ(KZS)+DZ(KZS-1))
 ISN 0020           DZK1=0.5*(DZ(KZS)+DZ(KZS+1))
 ISN 0021           DTQIJ=DT*QB(I,J)
 ISN 0022           CP1(I,J,KZS)= CP1(I,J,KZS)+DTQIJ*(1.-DZSK)/DZK
 ISN 0023           CP1(I,J,KZS+1)=CP1(I,J,KZS+1)+DTQIJ*DZSK/DZK1
 ISN 0024       100 CONTINUE
 ISN 0025       200 CONTINUE
              C
              C     ----------------- POINT SOURCE --------------------
              C
              C.... POINT SOURCE IS ADDED AT EFFECTIVE HEIGHT ZPR=ZP+ZR AND
              C       EFFECTIVE DOWNWIND LOCATION (X=X+U*DT,Y=Y+V*DT)
              C
 ISN 0026           IF (LSOUS(2) .EQ. 0) RETURN
              C
```

```
ISN 0028          LTMIN=LPRINT
ISN 0029          IF (ITM .NE.(ITM/LTMIN*LTMIN)) GO TO 410
            C
            C.... CALCULATE POINT SOURCE LOCATION IN GRID UNIT (XP,YP).
            C
ISN 0031           CALL     XYUTM1     (XPUTM,YPUTM,LM,XP,YP,LM,1)
            C
            C.... CALCULATE HEIGHT (ZP) IN GRID UNIT & STORE IN ZPR TEMPORARILY.
            C
ISN 0032          CALL     ZZGRID     (ZP,ZPR,Z,LM,LM,KM)
            C
            C *** COMPUTE EFFECTIVE HEIGHT & DOWNWIND LOCATION
            C
ISN 0033          DO 400 L=1,LM
ISN 0034          IXP=XP(L)+1.00001
ISN 0035          IYP=YP(L)+1.00001
ISN 0036          IZP=ZPR(L)+1.00001
ISN 0037          IF (IZP .GE.KM) IZP=KM
ISN 0039          KK=IZP
ISN 0040          IF(KK .GT. KN) KK=KN
ISN 0042          UL=U(IXP,IYP,KK)*UZF(IZP)
ISN 0043          VL=V(IXP,IYP,KK)*VZF(IZP)
ISN 0044          UIJK=SQRT(UL*UL+VL*VL)
ISN 0045          ZRL=ZR(L)
ISN 0046          IF ( ZR(L) .GT. 9000.) ZRL=PQB(L)*ZRPQ
ISN 0048          ZRR=ZRISE*ZRL
ISN 0049          ZPRL=ZP(L)+ZRR/(UIJK+EXP(-UIJK))**0.75
ISN 0050          ZPR(L)=ZPRL
            C.... TRANSFER TO NEW LOCATION DOWNWIND (EK,FK).
ISN 0051          TADVEC=DT
ISN 0052          IF (LPQ .GE. 3) GO TO 320
            C ... CT1=1./(16.*ALOG(10.))
ISN 0054          T1=CT1*DX(IXP)*DY(IYP)/(AKH(IZP)*AKF(IZP)*AKZ(IXP,IYP))
ISN 0055          AKFMEN=0.0
ISN 0056          DO 300 KZ=1,IZP
ISN 0057      300 AKFMEN=AKFMEN+AKF(KZ)
ISN 0058          AKFMEN=AKFMEN/IZP
ISN 0059          T2=0.25*ZPRL*ZPRL/(AKFMEN*AKZ(IXP,IYP))
            C     TADVEC=MINIMUM(T1,T2)
ISN 0060          TADVEC=T1
ISN 0061          IF (TADVEC.GT.T2) TADVEC=T2
ISN 0063      320 CONTINUE
ISN 0064          EK(L)=XPUTM(L)+0.001*TADVEC*UL
ISN 0065          FK(L)=YPUTM(L)+0.001*TADVEC*VL
ISN 0066      400 CONTINUE
            C
            C.... CONVERT TEMPORARY LOCATION (EK,FK) TO GRID UNIT (XP,YP).
            C
ISN 0067          CALL     XYUTM1     (EK,FK,LM,XP,YP,LM,1)
            C
            C.... CONVERT ZPR TO GRID UNIT.
            C
ISN 0068          CALL     ZZGRID     (ZPR,ZPR,Z,LM,LM,KM)
ISN 0069      410 CONTINUE
            C
            C.... NOW (XP,YP,ZPR) ARE IN GRID UNIT....
            C *** CONVERT EMISSION RATE TO DENSITY ACCORDING TO DT & GRID VOLUME ***
            C
```

```
ISN 0070          DO 600 L=1,LM
ISN 0071          IF (ZP(L) .GE. Z(KM)) GO TO 600
ISN 0073          IXP=XP(L)+1.001
ISN 0074          IYP=YP(L)+1.001
            C
            C.... DISCARD POINTS OUTSIDE COMPUTATIONAL REGION.
ISN 0075          IF ((IXP.LT. 1).OR.(IXP.GT.IM)) GO TO 600
ISN 0077          IF ((IYP.LT. 1).OR.(IYP.GT.JM)) GO TO 600
ISN 0079          ZPRL=ZPR(L)+1.001
ISN 0080          IZP=ZPRL
ISN 0081          DZP=ZPRL-IZP
ISN 0082          IZP1=IZP+1
ISN 0083          IF (IZP .LT. KM) GO TO 510
ISN 0085          IZP=KM
ISN 0086          IZP1=KM
ISN 0087          DZP=0.0
ISN 0088      510 CONTINUE
ISN 0089          DZK=0.5*DZ(IZP)
ISN 0090          IF (IZP.GT.1) DZK=0.5*(DZ(IZP)+DZ(IZP-1))
ISN 0092          DZK1=0.5*(DZ(IZP)+DZ(IZP1))
ISN 0093          DTPQ=DT*PQB(L)*QMUGM/(DX(IXP)*DY(IYP))
            C
            C *** ADD POINT SOURCE INTO THE GRID CELL ***
ISN 0094          IF ((LPQ .NE.1).AND.(LPQ.NE.3)) GO TO 500
            C
            C.... ADD POINT SOURCE AT ONE GRID ONLY
ISN 0096          CP1(IXP,IYP,IZP)=CP1(IXP,IYP,IZP)+DTPQ*(1.-DZP)/DZK
ISN 0097          CP1(IXP,IYP,IZP1)=CP1(IXP,IYP,IZP1)+DTPQ*DZP/DZK1
ISN 0098          GO TO 600
            C
            C.... DISTRIBUTE SOURCE AT FOUR NEIGHBORING GRIDS
ISN 0099      500 CONTINUE
ISN 0100          IF ((LPQ.NE.2).AND.(LPQ.NE.4)) GO TO 600
ISN 0102          A=XP(L)+0.5-IXP
ISN 0103          B=YP(L)+0.5-IYP
ISN 0104          DO 550 K=IZP,IZP1
ISN 0105          DTQ=DTPQ*(1.0-DZP)
ISN 0106          IF (K .EQ.IZP1) DTQ=DTPQ*DZP
ISN 0108          CP1(IXP,IYP,K)=CP1(IXP,IYP,K)+(1.0-A)*(1.0-B)*DTQ
ISN 0109          IF (IXP .GE. IM) GO TO 550
ISN 0111          CP1(IXP+1,IYP,K)= CP1(IXP+1,IYP,K)+ A*(1.0-B)*DTQ
ISN 0112          IF (IYP .GE. JM) GO TO 550
ISN 0114          CP1(IXP+1,IYP+1,K)= CP1(IXP+1,IYP+1,K)+ A*B*DTQ
ISN 0115          CP1(IXP  ,IYP+1,K)= CP1(IXP  ,IYP+1,K)+ (1.0-A)*B*DTQ
ISN 0116      550 CONTINUE
ISN 0117      600 CONTINUE
            C
ISN 0118          RETURN
            C     DEBUG SUBCHK
ISN 0119          END

*OPTIONS IN EFFECT*     NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*     SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =     118 ,PROGRAM SIZE =     4190

*STATISTICS*  NO  DIAGNOSTICS GENERATED
```

```
 LEVEL 21.6 ( MAY 72 )                                   OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                 C
   ISN 0002            SUBROUTINE SOUSIN (QB,ZS,Z0,PQB,XP,YP,ZP,ZR,IM,JM,LM)
                 C
                 C ... IT READS IN SO2 SOURCE FROM AREA AND POINT SOURCES   ...
                 C
   ISN 0003            DIMENSION
                    *          QB(IM,JM), ZS(IM,JM)   ,Z0(IM,JM)
                    *         ,ZR(LM), PQB(LM), XP(LM), YP(LM), ZP(LM)
                 C
   ISN 0004            COMMON    /AADATA/
                    *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                    *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                    *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                    *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                    *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                    *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                 C
   ISN 0005        870 FORMAT (' *END   IN READ SOUSIN* ')
   ISN 0006        880 FORMAT (' *ERROR IN READ SOUSIN* ')
                 C
                 C *** SEARCH RIGHT RECORD IN UNIT=KUNITS ***
   ISN 0007            IF (ITM .EQ. 0) REWIND KUNITS
   ISN 0009        100 CONTINUE
   ISN 0010            READ  (KUNITS,END=600,ERR  =700) KYR,KMO,KDAY,KHR
   ISN 0011            KHR2=KHR-2
   ISN 0012            IF ((KDAY .NE. IDAY) .OR.(KHR2 .NE. IHR)) GO TO 100
   ISN 0014            BACKSPACE KUNITS
                 C
                 C *** READ IN TIME, SOURCE, PLUME RISE, TOTAL SOURCE ***
   ISN 0015            READ  (KUNITS,END=700,ERR  =600) KYR,KMO,KDAY,KHR
                    &     ,QB,PQB,ZR,QBTOT,PQBTOT
   ISN 0016            PARM(6)=QBTOT
   ISN 0017            PARM(7)=PQBTOT
   ISN 0018            GO TO 9000
                 C
   ISN 0019        600 WRITE (JUNIT,870)
   ISN 0020            GO TO 9000
   ISN 0021        700 WRITE (JUNIT,880)
                 C
   ISN 0022       9000 RETURN
                 C     DEBUG SUBCHK
   ISN 0023            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       22 ,PROGRAM SIZE        998

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              121K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
               C
  ISN 0002              SUBROUTINE STABIT (V,ESKY,IHR,STAB,ITRAT)
               C
               C.... THIS SUBROUTINE ESTIMATES CONTINUOUS STABILITY CLASSES.
               C
  ISN 0003              DIMENSION W(5),F(5),START(6,5),X1(5),X2(5),Y3(5),Y4(5)
  ISN 0004              DIMENSION ISOLAR(24)
               C
  ISN 0005              DATA
                     *   W/2.0,3.0,5.0,6.0,8.0/,  F/2.0,1.0,2.0,1.0,2.0/
                     * ,ISOLAR/1,1,1,1,1,1,1,2,2,3,4,4,4,3,3,2,2,2,1,1,1,1,1,1/
                     * ,X1/2.0,2.5,2.5,3.1,3.2/,  X2/3.0,3.2,3.2,3.3,3.4/
                     * ,Y3/4.0,4.0,3.8,3.7,3.6/,  Y4/5.0,4.5,4.0,3.9,3.7/
                     * ,START/0.0,0.5,1.3,2.0,2.5,3.2,0.5,1.3,1.5,2.5,3.2,3.4,1.0,1.5,
                     * 2.5,3.0,3.3,3.5,5.0,5.0,4.1,3.8,3.6,3.5,6.0,6.0,5.1,4.0,3.8,3.6/
               C
               C.... STAB  = CONTINUOUS STABILITY CLASS, NEUTRAL CONDITION IS 3.5
               C     ISKY  = CLOUD COVER (=ESKY)
               C     V     = AVERAGE WIND SPEED (M/SEC)
               C     W     = END VALUES OF WIND SPEED RANGE
               C     F     = VALUE OF WIND SPEED RANGE INTERVAL
               C     INDEXW= INDEX OF WIND SPEED RANGE
               C     ISOLAR= INDEX OF INSOLATION AS TIME OF A DAY, =1 NIGHT TIME
               C              =2 TRANSSTION PERIOD; =3 SLIGHT INSOLATION;
               C              =4 MODERATE INSOLATION; =5 STRONG INSOLATION
               C           (CURRENT VALUE GIVEN IN DATA ARE FOR MONTH OF FEBRUARY)
               C     X1,X2 = STABILITY INDEX OF TRANSITION PERIOD FOR DAY TO NIGHT
               C     Y3,Y4 = SAME AS X1,X2 FOR NIGHT TO DAY
               C     START(INDEXW,IRADN) = STABILITY INDEX FOR NIGHT AND DAY TIME.
               C     IRADN = INDEX OF SKY CONDITION
               C              =1 STRONG INSOLATION; =2 MODERATE; =3 SLIGHT
               C              =4 CLOUDY NIGHT; =5 CLEAR NIGHT
               C
               C     ------------- TAPBE OF INDICES ---------------
               C     K=INDEXW :   1     2     3     4     5     6
               C     W(K)     :  2.0   3.0   5.0   6.0   8.0   ...
               C     'DAY & NIGHT TIME'
               C     START(K,N)
               C         N=1  :  0.0   0.5   1.3   2.0   2.5   3.2
               C         N=2  :  0.5   1.3   1.5   2.5   3.2   3.4
               C         N=3  :  1.0   1.5   2.5   3.0   3.3   3.5
               C         N=4  :  5.0   5.0   4.1   3.8   3.6   3.5
               C         N=5  :  6.0   6.0   5.1   4.0   3.8   3.6
               C     'DAY TO NIGHT'
               C       X1(K)  :  2.0   2.5   2.5   3.1   3.2   ...
               C       X2(K)  :  3.0   3.2   3.2   3.3   3.4   ...
               C     'NIGHT TO DAY'
               C       Y3(K)  :  4.0   4.0   3.8   3.7   3.6   ...
               C       Y4(K)  :  5.0   4.5   4.0   3.9   3.7   ...
               C
  ISN 0006              ISKY=ESKY
  ISN 0007              IH=IHR+1
               C
               C *** FIX INDEX OF WIND SPEED RANGE ***
  ISN 0008              DO 100 K=1,5
  ISN 0009              INDEXW=K
```

```
ISN 0010          P=V-W(K)
ISN 0011          IF (P.LT.0.0) GO TO 120
ISN 0013      100 CONTINUE
ISN 0014          INDEXW=6
ISN 0015      120 CONTINUE
            C
            C *** FIX INDEX OF SKY CONDITION ***
ISN 0016          NHR=ISOLAR(IH)
ISN 0017          GO TO (210,400,230,240,250),NHR
            C
            C.... INDEX FOR NIGHT TIME SKY CONDITION
ISN 0018      210 IRADN=5
ISN 0019          IF (ISKY.GT.4) IRADN=4
ISN 0021          GO TO 300
            C
            C.... INDEX FOR DAY TIME SKY CONDITION
ISN 0022      230 IRADN=3
ISN 0023          GO TO 300
ISN 0024      240 IRADN=2
ISN 0025          IF (ISKY.GT.4) IRADN=3
ISN 0027          GO TO 300
ISN 0028      250 IRADN=1
ISN 0029          IF (ISKY.GT.3) IRADN=2
ISN 0031          IF (ISKY.GT.7) IRADN=3
            C
            C.... SET ITRAT=0, CURRENT HOUR IS NOT IN TRANSITION PERIOD....
ISN 0033      300 CONTINUE
ISN 0034          ITRAT=0
            C
            C      ----------------- DAY OR NIGHT TIME STABILITY -----------------.
            C
ISN 0035          IF (INDEXW.GE.6) GO TO 320
ISN 0037          Q=START(INDEXW,IRADN)
ISN 0038          QP=START(INDEXW+1,IRADN)
ISN 0039          P=V-(W(INDEXW)-F(INDEXW))
ISN 0040          D=QP-Q
ISN 0041          STAB=Q+D*P/F(INDEXW)
ISN 0042          GO TO 900
ISN 0043      320 CONTINUE
ISN 0044          STAB=START(6,IRADN)
ISN 0045          GO TO 900
            C
            C     ------------------ TRANSITION PERIOD STABILITY ------------------
            C
ISN 0046      400 CONTINUE
ISN 0047          IF (INDEXW.GE.6) GO TO 500
ISN 0049          P=V-(W(INDEXW)-F(INDEXW))
            C.... IF ITRAT NOT EQUAL TO 0, PREVIOUS HOUR ALSO IN TRANSITION PERIOD
ISN 0050          IF (ITRAT.NE.1) GO TO 440
            C
            C *** FIRST HOUR OF TRANSIENT PERIOD ***
ISN 0052          IF (IH .LE. 12) GO TO 420
            C
            C ... DAY TO NIGHT
ISN 0054          Q=AMAX1(STAB,X1(INDEXW))
ISN 0055          QP=X2(INDEXW)
ISN 0056          IF (QP.LE.Q) GO TO 410
ISN 0058          D=QP-Q
```

```
ISN 0059          STAB=Q+D*P/F(INDEXW)
ISN 0060          GO TO 700
ISN 0061      410 D=3.5-Q
ISN 0062          STAB=Q+D*P/F(INDEXW)
ISN 0063          GO TO 700
            C
            C ... NIGHT TO DAY
ISN 0064      420 CONTINUE
ISN 0065          Q=AMIN1(STAB,Y4(INDEXW))
ISN 0066          QP=Y3(INDEXW)
ISN 0067          IF (QP.GE.Q) GO TO 430
ISN 0069          D=Q-QP
ISN 0070          STAB=QP+D*P/F(INDEXW)
ISN 0071          GO TO 700
ISN 0072      430 D=QP-3.5
ISN 0073          STAB=3.5+D*P/F(INDEXW)
ISN 0074          GO TO 700
            C
            C *** SECOND AND THIRD HOURS OF TRANSIENT PERIOD ***
ISN 0075      440 CONTINUE
ISN 0076          IF (IH.LT.12) GO TO 450
            C ... DAY TO NIGHT
ISN 0078          Q=AMAX1(STAB,X2(INDEXW))
ISN 0079          QP=Y3(INDEXW)
ISN 0080          D=QP-Q
ISN 0081          STAB=Q+D*P/F(INDEXW)
ISN 0082          GO TO 700
            C ... NIGHT TO DAY
ISN 0083      450 CONTINUE
ISN 0084          Q=X2(INDEXW)
ISN 0085          QP=AMIN1(STAB,Y3(INDEXW))
ISN 0086          D=QP-Q
ISN 0087          STAB=Q+D*P/F(INDEXW)
ISN 0088          GO TO 700
ISN 0089      500 STAB=3.5
            C
            C.... CURRENT HOUR IS IN TRANSITION PERIOD, SET (ITRAT.NE.0)
ISN 0090      700 CONTINUE
ISN 0091          ITRAT=99
            C
ISN 0092      900 CONTINUE
ISN 0093          RETURN
            CC.     DEBUG INIT
            C       DEBUG SUBCHK
ISN 0094          END
*OPTIONS IN EFFECT*      NAME= MAIN,OPT=02,LINECNT=60,SIZE=0000K,
*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
*STATISTICS*    SOURCE STATEMENTS =      93 ,PROGRAM SIZE =      1372
*STATISTICS*  NO DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                 113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                    C
  ISN 0002                SUBROUTINE STNCON (CC,IOBS,ICAL,KOBS,KCAL,ITOBS,NS,NSX
                         *                      ,IM,JM,XS,YS,JXS,JYS)
                    C
                    C.... THIS SUBROUTINE COMPUTES CONCENTRATION VALUES AT RAMS
                    C     STATION. IT ALSO COMPUTES SPATIAL AND TIME AVERAGE FOR
                    C     COMPUTED AND OBSERVED SO2 CONCENTRATIONS.
                    C
  ISN 0003                DIMENSION
                         *         CC(IM,JM),IOBS(NSX),ICAL(NSX),KOBS(NSX)
                         *         ,KCAL(NSX),ITOBS(NSX)
                         *         ,XS(NS),YS(NS),JXS(NS),JYS(NS)
                    C
  ISN 0004                COMMON    /AADATA/
                         *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                         *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                         *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                         *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                         *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,OCMIN
                         *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                    C
  ISN 0005                N=0
  ISN 0006                IOBS(NSX)=0
  ISN 0007                ICAL(NSX)=0
  ISN 0008                DO 200 L=1,NS
  ISN 0009                IOBS(L)=RAMS(5,L)+0.5
  ISN 0010                IXS=JXS(L)
  ISN 0011                IYS=JYS(L)
                    C
  ISN 0012                IF ((IXS.GT.1).AND.(IXS.LT.IM)) GO TO 100
  ISN 0014                ICAL(L)=0.5*(CC(IXS,IYS)+CC(IXS,IYS+1))
  ISN 0015                GO TO 110
  ISN 0016            100 CONTINUE
  ISN 0017                IF ((IYS.GT.1).AND.(IYS.LT.JM)) GO TO 120
  ISN 0019                ICAL(L)=0.5*(CC(IXS,IYS)+CC(IXS+1,IYS))
  ISN 0020            110 CONTINUE
  ISN 0021                IF (IOBS(L).GT.5000) GO TO 200
  ISN 0023                GO TO 130
                    C.... BI-LINEAR INTERPOLATION
  ISN 0024            120 CONTINUE
  ISN 0025                P=XS(L)-IXS+0.5
  ISN 0026                Q=YS(L)-IYS+0.5
  ISN 0027                IF (P.GE.0.) GO TO 122
  ISN 0029                P=P+1.0
  ISN 0030                IXS=IXS-1
  ISN 0031            122 CONTINUE
  ISN 0032                IF (Q.GE.0.) GO TO 124
  ISN 0034                Q=Q+1.0
  ISN 0035                IYS=IYS-1
  ISN 0036            124 CONTINUE
  ISN 0037                R=(1.-P)*(1.-Q)*CC(IXS,IYS)+P*(1.-Q)*CC(IXS+1,IYS)
                         *      +Q*(1.-P)*CC(IXS,IYS+1)+P*Q*CC(IXS+1,IYS+1)
  ISN 0038                ICAL(L)=R+0.5
  ISN 0039                IF (IOBS(L).GT.5000) GO TO 200
                    C
                    C.... SPATIAL AVERAGE
```

```
ISN 0041          IOBS(NSX)=IOBS(NSX)+IOBS(L)
ISN 0042          ICAL(NSX)=ICAL(NSX)+ICAL(L)
ISN 0043          N=N+1
ISN 0044     130 CONTINUE
            C
            C.... 24 HOUR AVERAGE
ISN 0045          KOBS(L)=KOBS(L)+IOBS(L)
ISN 0046          KCAL(L)=KCAL(L)+ICAL(L)
ISN 0047          ITOBS(L)=ITOBS(L)+1
ISN 0048     200 CONTINUE
ISN 0049          IF(N .EQ. 0) GO TO 210
ISN 0051          IOBS(NSX)=IOBS(NSX)/N
ISN 0052          ICAL(NSX)=ICAL(NSX)/N
ISN 0053          KOBS(NSX)=KOBS(NSX)+IOBS(NSX)
ISN 0054          KCAL(NSX)=KCAL(NSX)+ICAL(NSX)
ISN 0055          ITOBS(NSX)=ITOBS(NSX)+1
ISN 0056     210 CONTINUE
            C
ISN 0057          IF (IHR.NE.0) RETURN
            C
ISN 0059          DO 300 L=1,NSX
ISN 0060          KOBS(L)=KOBS(L)/ITOBS(L)
ISN 0061          KCAL(L)=KCAL(L)/ITOBS(L)
ISN 0062     300 CONTINUE
            C
ISN 0063          RETURN
            C     DEBUG SUBCHK
ISN 0064          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       63 ,PROGRAM SIZE =      1696

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                          113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                   C
  ISN 0002               SUBROUTINE TIMEX (NMONDY)
                   C
                   C.... THIS ROUTINE FIX INDICES OF SIMULATION AND REAL TIME.
                   C
  ISN 0003               DIMENSION NMONDY(12)
  ISN 0004               DATA NDAYHR/24/,NHRSEC/3600/
                   C
  ISN 0005               COMMON    /AADATA/
                      *    IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                      *   ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                      *   ,LPRINT,LTSTOP,LTSOUS,LTWIND
                      *   ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                      *   ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                      *   ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                   C
                   C *** FIX SIMULATION TIME INDICIES ***
                   C
  ISN 0006               TM=TM+DT
  ISN 0007               ITM=TM
  ISN 0008               ITSTEP=ITSTEP+1
  ISN 0009               ITMHR=ITM/NHRSEC*NHRSEC
  ISN 0010               TSEC=TM-ITMHR
  ISN 0011               ITSEC=TSEC
                   C
                   C *** FIX INDICIES OF MONTH, DAY AND HOUR ***
                   C
  ISN 0012               IF (ITM .NE.  ITMHR) GO TO 110
  ISN 0014               ITOTHR=ITOTHR+1
  ISN 0015               IHR=IHR+1
  ISN 0016               IF ( IHR .LT. NDAYHR) GO TO 110
  ISN 0018               IDAY= IDAY+1
  ISN 0019               IHR= 0
  ISN 0020               IF ( IDAY .LE. NMONDY(IMO)) GO TO 110
  ISN 0022               IMO=IMO+1
  ISN 0023               IDAY=1
  ISN 0024               IF ( IMO .LE. 12) GO TO 110
  ISN 0026               IYR=IYR+1
  ISN 0027               IMO=1
  ISN 0028           110 CONTINUE
                   C
  ISN 0029               RETURN
                   C     DEBUG SUBCHK
  ISN 0030               END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       29 ,PROGRAM SIZE        514

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                               121K BYTES OF CORE NOT USED
```

LEVEL 21.6 (MAY 72)   OS/360 FORTRAN H

```
        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                 C
  ISN 0002             SUBROUTINE UVFLUX ( CP1,  C,U,UZF,DX,DY,RDX,RDY,Z,EK,FK,
                     *                         IM,JM,KM,IMJM,IJKM,LX,LY,IJKN)
                 C
                 C.... THIS SUBROUTINE COMPUTES THE HORIZONTAL ADVECTION TERMS OF
                 C....  CONCENTRATION EQUATION.
                 C.... THIS PROGRAM CAN BE USED FOR X AND Y ADVECTION BY INTERCHANGING
                 C....  VARIABLES ARGUMENTS IN CALL STATEMENT.
                 C.... THE SECOND ORDER CENTRAL FINITE DIFFERENCE SCHEME IS USED.
                 C.... THE 3-D VARIABLE A(I,J,K) IS REPRESENTED BY VECTOR A(IJK).
                 C
  ISN 0003             DIMENSION
                     *        CP1(IJKM),C(IJKM),U(IJKN),UZF(KM)
                     *     ,DX(IM),DY(JM),RDX(IM),RDY(JM),Z(KM)
                     *     ,EK(IM) ,FK(IM),DTDXI(40)
                 C
  ISN 0004             COMMON    /AADATA/
                     *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                     *  ,IYR,IMO,IDAY,IHR,ITM,ITMMR,ITSEC,ITOTMR,ITSTEP,DT,TM,TSEC
                     *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                     *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                     *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                     *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                 C
                 C *** COEFFICIENTS OF FINITE DIFF. FORMULA ***
  ISN 0005             IM11=IM-1
  ISN 0006             JM11=JM-1
  ISN 0007             EK(1)=0.5
  ISN 0008             FK(IM)=0.5
  ISN 0009             DO 10 I=2,IM
  ISN 0010             FK(I-1)=1.0/(1.0+RDX(I))
  ISN 0011          10 EK(I)= RDX(I)*FK(I-1)
  ISN 0012             KK=0
  ISN 0013             DO 30 I=1,IM
  ISN 0014          30 DTDXI(I)=DT/DX(I)
                 C
                 C *** COMPUTE ADVECTION ***
  ISN 0015             KN=IJKN/IMJM
  ISN 0016             KN1=KN+1
  ISN 0017             KKVMAX=(KN-1)*IMJM
  ISN 0018             KKV=-IMJM
                 C
  ISN 0019             DO 500 K= 2,KM
  ISN 0020             KK= KK+ IMJM
  ISN 0021             KKV=KKV+IMJM
                 C     FOR K GT KN1, THE WIND ARE THE SAME AS K=KN1.
  ISN 0022             IF ( K .GT. KN1) KKV=KKVMAX
  ISN 0024             ZK= Z(K)
                 C
                 C *** PROCESS INTERIOR GRID POINTS ***
  ISN 0025             JJ=-LY
  ISN 0026             DO 200 J= 1,JM
  ISN 0027             JJ= JJ+LY
  ISN 0028             JLY=JJ
  ISN 0029             ILX= -LX
  ISN 0030             DO 150 I= 1,IM11
```

```
ISN 0031          ILX= ILX+ LX
ISN 0032          IJ= 1+ ILX+JLY
ISN 0033          IJK = IJ+KK
ISN 0034          IJKV=IJ+KKV
            C     ALF= DTDX*(U(IJKV)+U(IJKV+LX))
ISN 0035          ALF= DTDXI(I)*U(IJKV)*UZF(K)
ISN 0036          AF1= ALF
ISN 0037          A3= AF1*AF1
ISN 0038          CIJK= C(IJK)
ISN 0039          CIJK1= C(IJK+LX)
ISN 0040          CPIJK =CP1(IJK)
ISN 0041          CPIJK1=CP1(IJK+LX)
ISN 0042          T1= CIJK+CIJK1
ISN 0043          T3= CIJK-CIJK1
            C.... UF IS FLUX FROM ONE GRID POINT TO ANOTHER.
ISN 0044          UF=AF1*T1+A3*T3
ISN 0045          CCIJK =CPIJK -UF*EK(I)
ISN 0046          CCIJK1=CPIJK1+UF*FK(I)
            C.... CHECK TOTAL FLUX IS NOT EXCEED THE MASS IN UPWIND GRID
            C....  CELL. IF EXCEEDED, READJUST THE FLUX TO PREVENT THE
            C....  VALUES OF C TO BECOME NEGATIVE DUE TO TRUNCATION ERROR.
ISN 0047          IF ((CCIJK .GE. 0.0) .AND. (CCIJK1 .GE. 0.0)) GO TO 110
ISN 0049          IF ( CCIJK .LT. 0.) UF=CPIJK/EK(I)
ISN 0051          IF ( CCIJK1 .LT. 0.0) UF=-CPIJK1/FK(I)
ISN 0053          CCIJK =CPIJK -UF*EK(I)
ISN 0054          CCIJK1=CPIJK1+UF*FK(I)
ISN 0055      110 CP1(IJK)=CCIJK
ISN 0056          CP1(IJK+LX)=CCIJK1
ISN 0057      150 CONTINUE
ISN 0058      200 CONTINUE
            C
            C *** PROCESS INFLOW AND OUTFLOW BOUNDARY GRID POINTS ***
            C
ISN 0059          DO 400 I= 1, IM, IM11
ISN 0060          ILX= (I-1)*LX
ISN 0061          JLY= -LY
ISN 0062          DO 350 J= 1, JM
ISN 0063          JLY= JLY+ LY
ISN 0064          IJ= 1+ ILX+JLY
ISN 0065          IJK = IJ+KK
ISN 0066          IJKV=IJK-IMJM
ISN 0067          IF ( K .GT. KN1) IJKV=IJ+KKVMAX
ISN 0069          IF ( I .EQ. IM) GO TO 320
            C     ALF= DTDX*(U(IJKV)+U(IJKV+LX))
ISN 0071          ALF=DTDXI(I)*U(IJKV)*UZF(K)
ISN 0072          IF ( ALF .GT. 0.0 ) GO TO 330
ISN 0074          CIJK=C(IJK)
ISN 0075          CIJK1=C(IJK+LX)
ISN 0076          T3=CIJK-CIJK1
ISN 0077          UF= ALF*T3
ISN 0078          IF (CIJK1 .LT. UF) UF=CIJK1
ISN 0080          CP1(IJK)= CIJK+UF
ISN 0081          GO TO 330
ISN 0082      320 CONTINUE
            C     ALF= DTDX*(U(IJKV)+U(IJKV-LX))
ISN 0083          ALF=DTDXI(I)*U(IJKV)*UZF(K)
ISN 0084          IF ( ALF .LT. 0.0 ) GO TO 330
ISN 0086          CIJK=C(IJK)
```

```
ISN 0087            CIJK1=C(IJK-LX)
ISN 0088            T3=CIJK1-CIJK
ISN 0089            UF= ALF*T3
ISN 0090            IF (CIJK1 .LT. UF) UF=CIJK1
           C        OUTFLOW BOUNDRY ,FIRST ORDER UPSTREAM SCHEME IS USED .....
           C        NOTE THAT CIJK IS OLD TIME C ...
ISN 0092            CP1(IJK)= CIJK+UF
ISN 0093        330 CONTINUE
ISN 0094            IF ( CP1(IJK) .LT. 0.0) CP1(IJK)=0.0
ISN 0096        350 CONTINUE
ISN 0097        400 CONTINUE
           C
ISN 0098        500 CONTINUE
           C
ISN 0099            RETURN
           C        DEBUG SUBCHK
ISN 0100            END

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       99 ,PROGRAM SIZE =      2228

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              109K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                   C
  ISN 0002               SUBROUTINE UVINTP (UU,VV,IN,JN,U,V,IM,JM,DX,DY,DELX)
                   C
                   C.... THIS ROUTINE OBTAINS THE SURFACE WIND FIELD AT NEMERICAL
                   C      GRIDS BY INTERPOLATING ANALYZED FIELD AT WIND GRIDS.
                   C.... THE INTERPOLATION SCHEME IS BI-LINEAR.
                   C     DELX - DIMENSION OF WIND GRID.
                   C
  ISN 0003               DIMENSION UU(IN,JN),VV(IN,JN),U(IM,JM),V(IM,JM)
                        &          ,DX(IM),DY(JM)
                   C
  ISN 0004               IF ((IM.LT.IN).OR.(JM.LT.JN)) RETURN
                   C
  ISN 0006               DELXI=0.001/DELX
  ISN 0007               YJ=-DY(1)*0.5
  ISN 0008               DO 100  J=1,JM
  ISN 0009               YJ=YJ+DY(J)
  ISN 0010               YW=YJ*DELXI+1.0001
  ISN 0011               JW=YW
  ISN 0012               Q=YW-JW
  ISN 0013               Q1=1.-Q
  ISN 0014               XI=-DX(1)*0.5
  ISN 0015               DO 100 I=1,IM
  ISN 0016               XI=XI+DX(I)
  ISN 0017               XW=XI*DELXI+1.0001
  ISN 0018               IW=XW
  ISN 0019               P=XW-IW
  ISN 0020               P1=1.-P
  ISN 0021               D1=P1*Q1
  ISN 0022               D2= P*Q1
  ISN 0023               D3=P1* Q
  ISN 0024               D4= P* Q
  ISN 0025               U(I,J)=D1*UU(IW,JW)+D2*UU(IW+1,JW)+D3*UU(IW,JW+1)+D4*UU(IW+1,JW+1)
  ISN 0026               V(I,J)=D1*VV(IW,JW)+D2*VV(IW+1,JW)+D3*VV(IW,JW+1)+D4*VV(IW+1,JW+1)
  ISN 0027           100 CONTINUE
                   C
  ISN 0028               RETURN
                   C     DEBUG SUBCHK
  ISN 0029               END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       28 ,PROGRAM SIZE       1220

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                          121K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                    OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                     C
   ISN 0002                SUBROUTINE UVZF (U,V,U1,V1,UZF,VZF,WZF,Z,IM,JM,KM,KN,LX)
                     C
                     C.... THIS SUBROUTINE COMPUTES VERTICAL WIND PROFILE.
                     C
   ISN 0003                DIMENSION U(IM,JM,KN),V(IM,JM,KN),U1(IM,JM),V1(IM,JM)
   ISN 0004                DIMENSION Z(KM),UZF(KM),VZF(KM),WZF(KM)
                     C
   ISN 0005                COMMON    /AADATA/
                          *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                          *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                          *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                          *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                          *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                          *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                     C
                     C
                     C
                     C
                     C.... COMPUTE WIND POWER LAW CONSTANT AND WIND ANGLE CHANGE.
   ISN 0006                UVP=PARM(1)
   ISN 0007                THETAP=PARM(2)
   ISN 0008                P=PMIN
   ISN 0009                IF ((UVP.LE.0.5).OR.(UVP.GE.30.)) GO TO 10
   ISN 0011                P=ALOG(UVP/PARM(1))*0.8
   ISN 0012                IF ((PARM(1).GE.6.).AND.(P.GT.0.3)) P=0.3
   ISN 0014             10 CONTINUE
   ISN 0015                IF (P.LT.PMIN) P=PMIN
   ISN 0017                IF (P.GT.PMAX) P=PMAX
   ISN 0019                DTHETA=THETAP-PARM(2)
   ISN 0020                IF ((THETAP.GT.360.).OR.(PARM(2).GT.360.)) DTHETA=0.
   ISN 0022                IF (ABS(DTHETA).GT.180.) DTHETA=DTHETA-SIGN(1.,DTHETA)*360.
   ISN 0024                IF (ABS(DTHETA).GT.25.) DTHETA=25.*SIGN(1.,DTHETA)
                     C
                     C
   ISN 0026                GO TO (100,200,300),LWIND
                     C
                     C     --------------------------- LWIND=1 ---------------------------.
                     C.... LWIND=1, ALL UPPER WIND ABOVE HEIGHT HP ARE SAME AS OBSERED.
   ISN 0027            100 CONTINUE
   ISN 0028                AHPZS=1.0/ALOG(HP/HS)
   ISN 0029                DO 190 K=2,KN
   ISN 0030                ZK=Z(K+1)
   ISN 0031                IF (ZK.GE.HP) ZK=HP
   ISN 0033                DO 190 J=1,JM
   ISN 0034                DO 190 I=1,IM
   ISN 0035                UK=U1(I,J)
   ISN 0036                VK=V1(I,J)
   ISN 0037                IF( LX .EQ. 1) GO TO 110
   ISN 0039                UK=V1(I,J)
   ISN 0040                VK=U1(I,J)
   ISN 0041            110 CONTINUE
                     C
   ISN 0042                CALL       WINDER      ( UK,VK,UV,THETA,1)
   ISN 0043                IF ( UV .EQ.  0.0) GO TO 130
   ISN 0045                IF ( UVP .LE. UV) GO TO 150
```

```
ISN 0047          P   ALOG( UVP/UV )*AHPZS
ISN 0048          UVZ= UV*(ZK/HS)**P
ISN 0049          DTHETA= THETAP- THETA
            C.... AVOID DTHET GT 180. DEG
ISN 0050          IF ( ABS (DTHETA) .GT. 180.) DTHETA=DTHETA-SIGN(1.,DTHETA)*360.
ISN 0052          THETAZ= THETAP-(HP-ZK)/(HP- HS)* DTHETA
            C
ISN 0053      120 CALL       WINDER     (UK,VK,UVZ,THETAZ, 2)
ISN 0054          IF ( LX .EQ. 1) GO TO 150
ISN 0056          UX= UK
ISN 0057          UK= VK
ISN 0058          VK= UX
ISN 0059          GO TO 150
            C.... IF UV=0., PARABOLIC PROFILE IS ASSUMED
ISN 0060      130 ZPS= (ZK-HS)/(HP-HS)
ISN 0061          UVZ= UVP*(1.0- ZPS*ZPS)
ISN 0062          THETAZ= THETAP
ISN 0063          GO TO 120
ISN 0064      150 CONTINUE
ISN 0065          U(I,J,K)=UK
ISN 0066          V(I,J,K)=VK
ISN 0067      190 CONTINUE
ISN 0068          GO TO 400
            C
            C       ------------------------- LWIND=2 -------------------------.
            C
            C.... LWIND=2 , UPPER WIND HAS SAME DIRECTION AS SURFACE WIND.
            C       THE WIND SPEED ARE COMPUTED BY POWER LOW.
ISN 0069      200 CONTINUE
ISN 0070          IF(KN .GT. 1) GO TO 230
ISN 0072          DO 220 K=2,KM
ISN 0073          ZK=Z(K)
ISN 0074          IF(ZK .GT. HP) ZK=HP
ISN 0076          UZF(K)=(ZK/HS)**P
ISN 0077      220 CONTINUE
ISN 0078          GO TO 400
ISN 0079      230 CONTINUE
ISN 0080          DO 290 K=2,KN
ISN 0081          ZK=Z(K+1)
ISN 0082          ZKP=(ZK/HS)**P
ISN 0083          DO 290 J=1,JM
ISN 0084          DO 290 I=1,IM
ISN 0085          U(I,J,K)=U1(I,J)*ZKP
ISN 0086          V(I,J,K)=V1(I,J)*ZKP
ISN 0087      290 CONTINUE
ISN 0088          GO TO 400
            C
            C       ------------------------- LWIND=3 -------------------------.
            C
            C.... LWIND=3, UPPER WIND HAS EQUAL ANGLE CHANGE AS OBSERVED DATA.
ISN 0089      300 CONTINUE
ISN 0090          DO 390 K=2,KN
ISN 0091          ZK=Z(K+1)
ISN 0092          ZKP=(ZK/HS)**P
ISN 0093          ZKTHE=(ZK-HS)/(HP-HS)
ISN 0094          DO 390 J=1,JM
ISN 0095          DO 390 I=1,IM
ISN 0096          U1IJ=U1(I,J)
```

```
   ISN 0097          V1IJ=V1(I,J)                                               00004130
            C                                                                   00004140
   ISN 0098          CALL     WINDER     (U1IJ,V1IJ,UV1,THETA1,1)               00004150
   ISN 0099          UVZK=UV1*ZKP                                               00004160
   ISN 0100          THETAZ= THETA1+ ZKTHE*DTHETA                               00004170
            C                                                                   00004180
   ISN 0101          CALL     WINDER     (UK,VK,UVZK,THETAZ,2)                  00004190
   ISN 0102          U(I,J,K)= UK                                               00004200
   ISN 0103          V(I,J,K)= VK                                               00004210
   ISN 0104      390 CONTINUE                                                   00004220
   ISN 0105      400 CONTINUE                                                   00004230
            C                                                                   00004240
   ISN 0106          RETURN                                                     00004250
            C        DEBUG SUBCHK                                               00004260
   ISN 0107          END                                                        00004270

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      106 ,PROGRAM SIZE =      2706

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                   97K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
              C
  ISN 0002          SUBROUTINE WINDER ( U, V, UV, THETA, I)
              C
              C.... THIS SUBROUTINE CONVERTS WIND VECTOR TO COMPONENT OR
              C     WIND COMPONENTS TO VECTOR, U=E-W WIND,V=N-S WIND,
              C     UV=WIND SPEED, THETA=WIND DIRECTION.IN DEG....
              C.... I=1 CONVERT  U,V TO UV, THETA IN DEG
              C.... I=2 CONVERT UV,THETA TO U,V
              C
  ISN 0003          DATA PAIDEG /57.295776/
              C.... PAIDEG=180./3.1416
              C
  ISN 0004          IF ( I .GT. 1) GO TO 100
  ISN 0006          UV= SQRT (U*U+V*V)
  ISN 0007          IF (UV .EQ. 0.0) GO TO 20
  ISN 0009          THETA = ATAN2 (-U,-V)*PAIDEG
  ISN 0010          IF (THETA .LT. 0.0) THETA= THETA +360.0
  ISN 0012          RETURN
              C
  ISN 0013       20 THETA=0.0
  ISN 0014          RETURN
              C
  ISN 0015      100 ATHETA= THETA/PAIDEG
  ISN 0016          U= -UV*SIN(ATHETA)
  ISN 0017          V= -UV*COS(ATHETA)
              C
  ISN 0018          RETURN
              C     DEBUG SUBCHK
  ISN 0019          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      18 ,PROGRAM SIZE =      564

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                          125K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE WINDGR (U1,V1,UU,VV,NEAR,USTN,VSTN,XSUTM,YSUTM
                       &                  ,IUTM,JUTM,DX,DY,IM,JM,IN,JN,NS)
                  C
                  C.... THIS ROUTINE GENERATES WHOLE WIND FIELD FROM WIND DATA MEASURED
                  C     AT THE STATIONS.
                  C
  ISN 0003              DIMENSION U1(IM,JM),V1(IM,JM),UU(IN,JN),VV(IN,JN),NEAR(IN,JN)
                       &          ,USTN(NS),VSTN(NS),XSUTM(NS),YSUTM(NS)
                       &          ,IUTM(IM),JUTM(JM),DX(IM),DY(JM)
                  C
  ISN 0004              COMMON    /AADATA/
                       *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                       *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                       *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                       *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                       *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                       *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                  C
  ISN 0005              IF ((IN.GT.IM).OR.(JN.GT.JM)) RETURN
                  C
                  C *** KWIND = 3  UNIFORMED WIND FIELD.
  ISN 0007              IF (KWIND.NE.3) GO TO 200
  ISN 0009              UVS=PARM(1)
  ISN 0010              THS=PARM(2)
                  C
  ISN 0011              CALL     WINDER     (USTA,VSTA,UVS,THS,2)
                  C
  ISN 0012              DO 100 J=1,JM
  ISN 0013              DO 100 I=1,IM
  ISN 0014              U1(I,J)=USTA
  ISN 0015              V1(I,J)=VSTA
  ISN 0016          100 CONTINUE
  ISN 0017              RETURN
                  C
                  C *** KWIND = 4  VARIABLE WIND FIELD. ***
                  C.... WEIGHTED INTERPOLATION SCHEME IS USED.
  ISN 0018          200 CONTINUE
  ISN 0019              IF (KWIND.NE.4) RETURN
                  C
                  C.... SPECIFY WIND GRID, RADIUS OF INFLUENCE AND FIND A NEAREST
                  C      STATION TO WIND GRID (I,J)
  ISN 0021              IF (ITM.NE.0) GO TO 300
  ISN 0023              IDELX=(IUTM(IM)-IUTM(1)+0.001*DX(IM))/(IN-1)+0.1
  ISN 0024              DELX=IDELX
  ISN 0025              IDELY=(JUTM(JM)-JUTM(1)+0.001*DY(JM))/(JN-1)+0.1
  ISN 0026              DELY=IDELY
  ISN 0027              IF (DELX.LT.DELY) DELX=DELY
  ISN 0029              RTEST=(2.*DELX)**2
  ISN 0030              IRAD=2
  ISN 0031              DO 250 J=1,JN
  ISN 0032              DYB=(J-1)*DELX+JUTM(1)
  ISN 0033              DO 250 I=1,IN
  ISN 0034              DXB=(I-1)*DELX+IUTM(1)
  ISN 0035              SDIST=1200.
  ISN 0036              DO 230 K=1,NS
```

```
      DDX=DXB-XSUTM(K)
      DDY=DYB-YSUTM(K)
      DISTR=DDX*DDX+DDY*DDY
      IF (DISTR.GT.SDIST) GO TO 230
      SDIST=DISTR
      KK=K
  230 CONTINUE
      NEAR(I,J)=KK
  250 CONTINUE
C
C *** COMPUTE U AND V FROM OBSERVED UBAR AND THETA AT STATION.
  300 CONTINUE
      DO 310 K=1,NS
      UVS=RAMS(1,K)
      THS=RAMS(2,K)
      USTN(K)=99.9
      VSTN(K)=99.9
      IF ((UVS.GE.50.).OR.(THS.GT.360.)) GO TO 310
C
      CALL     WINDER     (USTA,VSTA,UVS,THS,2)
C
      USTN(K)=USTA
      VSTN(K)=VSTA
  310 CONTINUE
C
C *** OBTAIN INITIAL GUESS FIELD ***
      DO 400 J=1,JN
      DO 400 I=1,IN
      KK=NEAR(I,J)
      IF (USTN(KK).LT.50.) GO TO 380
      DXB=(I-1)*DELX+IUTM(1)
      DYB=(J-1)*DELX+JUTM(1)
      SDIST=1200.
      DO 350 K=1,NS
      DDX=DXB-XSUTM(K)
      DDY=DYB-YSUTM(K)
      DISTR=DDX*DDX+DDY*DDY
      IF (DISTR.GT.SDIST) GO TO 350
      IF (USTN(K).GT.50.) GO TO 350
      SDIST=DISTR
      KK=K
  350 CONTINUE
  380 CONTINUE
      U1(I,J)=USTN(KK)
      V1(I,J)=VSTN(KK)
  400 CONTINUE
C
C *** ITERATION ***
C
      DO 420 J=1,JN
      DO 420 I=1,IN
      D=0.
      USUM=0.
      VSUM=0.
      KIS=MAX0((I-IRAD), 1)
      KIF=MIN0((I+IRAD),IN)
      KJS=MAX0((J-IRAD), 1)
      KJF=MIN0((J+IRAD),JN)
```

```
   ISN 0091          DO 410 KJ=KJS,KJF
   ISN 0092          RY=(J-KJ)*(J-KJ)
   ISN 0093          DO 410 KI=KIS,KIF
   ISN 0094          RX=(I-KI)*(I-KI)
   ISN 0095          RDIST=RX+RY
   ISN 0096          IF ((RDIST.GT.RTEST).OR.(RDIST.LE.0.01)) GO TO 410
   ISN 0098          DISTSQ=1./RDIST
   ISN 0099          USUM=USUM+U1(KI,KJ)*DISTSQ
   ISN 0100          VSUM=VSUM+V1(KI,KJ)*DISTSQ
   ISN 0101          D=D+DISTSQ
   ISN 0102      410 CONTINUE
   ISN 0103          UU(I,J)=USUM/D
   ISN 0104          VV(I,J)=VSUM/D
   ISN 0105      420 CONTINUE
               C
               C *** OBTAIN WIND AT NUMERICAL GRIDS FROM UU AND VV
               C.... LINEAR INTERPOLATION IS USED.
               C
   ISN 0106          CALL     UVINTP     (UU,VV,IN,JN,U1,V1,IM,JM,DX,DY,DELX)
               C
   ISN 0107          RETURN
               C     DEBUG SUBCHK
               CC     DEBUG INIT (UU,VV,U1,V1,USTN,VSTN)
   ISN 0108          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      107 ,PROGRAM SIZE =      3452

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                 97K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                    OS/360  FORTRAN H

         COMPILER OPTIONS   NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE WINDIN ( U,V,W,U1,V1,UU,VV,NEAR,USTN,VSTN,UZF,VZF,WZF
                     *             ,XSUTM,YSUTM,IUTM,JUTM,Z,DX,DY,DZ
                     *             ,IM,JM,KM,IMJM,IJKN,IN,JN,KN,IS,NS)
                C
                C.... THIS ROUTINE READS IN SURFACE WIND FIELD AND RAMS DATA.
                C     IT CALLS ROUTINE UVZF AND WWZF TO CONSTRUCT 3-D WIND (U,V,W)
                C ... KWIND = 1 : INPUT OBJECTIVE ANALYZED WIND U1(IM,JM) & V1(IM,JM);
                C             2 : INPUT SUBJECTIVE ANALYZED WIND UU(IN,JN) & VV(IN,JN);
                C             3 ; INPUT RAMS DATA AND GENERATE UNIFORM WIND FIELD;
                C             4 : INPUT RAMS DATA AND GENERATE VARIABLE WIND FIELD.
                C.... RAMS(I,J) = DATA FROM RAMS STATION. J IS STATION INDEX.
                C                I=1, WIND SPEED; =2, WIND DIRECTION; =3, 1ST LEVEL TEMP,
                C                 =4, 2ND LEVEL TEMP; =5, SO2 CONC.; =6, RADIATION.
                C
  ISN 0003            DIMENSION
                     *        U(IM,JM,KN), V(IM,JM,KN) ,W(IM,JM,KN)
                     *       ,U1(IM,JM),V1(IM,JM),DX(JM),DY(JM),DZ(KM),Z(KM)
                     *       ,UU(IN,JN),VV(IN,JN),IS(NS),UZF(KM),VZF(KM),WZF(KM)
                     *       ,NEAR(IN,JN),XSUTM(NS),YSUTM(NS),IUTM(IM),JUTM(JM)
                     *       ,USTN(NS),VSTN(NS)
                C
  ISN 0004            COMMON    /AADATA/
                     *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                     *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                     *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                     *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                     *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                     *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                C
  ISN 0005        800 FORMAT (4I3/,(3(1X,F3.1,F3.0,2F4.1,F3.0,F4.0)))
  ISN 0006        870 FORMAT (' *END IN READ WIND DATA* ')
  ISN 0007        880 FORMAT (' *ERR IN READ WIND DATA* ')
                C
                C *** INPUT WIND AAD RAMS DATA ***
                C
                C
  ISN 0008            GO TO (100,200,300,300), KWIND
                C
                C.... KWIND   1, INPUT PREPROCESSED OBJECTIVE WIND U1,V1
  ISN 0009        100 READ (KUNITW,END=500,ERR=600) KYR,KMO,KDAY,KHR,U1,V1,RAMS
  ISN 0010            IF ((KDAY .NE. IDAY) .OR. (KHR .NE. IHR)) GO TO 100
  ISN 0012            GO TO 400
                C
                C.... KWIND   2, INPUT SUBJECTIVE ANALYZED WIND UU,VV
  ISN 0013        200 CONTINUE
  ISN 0014        210 READ (KUNITW,END=500,ERR=600) KYR,KMO,KDAY,KHR,UU,VV,RAMS
  ISN 0015            IF((KDAY .NE. IDAY).OR. (KHR .NE. IHR)) GO TO 210
                C ... INTERPOLATE UU(IN,JN) TO U(IM,JM) BY LINEAR INTERPOLATION
                C          (THIS PORTION OF PROGRAM NEEDS MODIFICATION DEPENDING ON THE
                C           GRID SYSTEM USED IN THE SUBJECTIVE ANALYSIS OF WIND FIELD)
  ISN 0017            DELX=IM/(IN-1)
  ISN 0018            DELY=JM/(JN-1)
  ISN 0019            IF (DELX.LT.DELY) DELX=DELY
  ISN 0021            CALL UVINTP (UU,VV,IN,JN,U1,V1,IM,JM,DX,DY,DELX)
  ISN 0022            GO TO 400
```

```
              C
              C.... KWIND = 3 & 4, INPUT RAMS DATA AND CALL ROUTINE WINDGR TO GENERATE
              C                    SURFACE WIND FIELD AT NUMERICAL GRID
ISN 0023        300 CONTINUE
ISN 0024            READ (KUNITW,800,END=500,ERR=600) KYR,KMO,KDAY,KHR,RAMS
ISN 0025            IF ((KDAY.NE.IDAY).OR.(KHR.NE.IHR)) GO TO 300
              C
              C     CALL      WINDGR      (U1,V1,UU,VV,NEAR,USTN,VSTN,XSUTM,YSUTM
              C    &                      ,IUTM,JUTM,DX,DY,IM,JM,IN,JN,NS)
              C
ISN 0027        400 CONTINUE
              C
              C *** OBTAIN AVERAGE WIND SPEED AND STABILITY INDEX ***
              C
ISN 0028            IF (ITM.GT.0) GO TO 410
ISN 0030            S=0.0
ISN 0031            ITRAT=0
ISN 0032        410 SSS=S+3.5
ISN 0033            TN=0.0
ISN 0034            UBAR=0.0
ISN 0035            DO 420 I=1,NS
ISN 0036            IF (RAMS(1,I).GT.50.) GO TO 420
ISN 0038            TN=TN+1.
ISN 0039            UBAR=UBAR+RAMS(1,I)
ISN 0040        420 CONTINUE
ISN 0041            UBAR=UBAR/TN
ISN 0042            ESKY=RAMS(6,1)
              C
ISN 0043            CALL      STABIT      (UBAR,ESKY,IHR,SSS,ITRAT)
ISN 0044            S=SSS-3.5
              C
              C.... STORE SUMMARY OF METEOROLOGICAL PARAMETERS ON ARRAY PARM
              C     THIS PART OF PROGRAM IS SUBJECTED TO BE MODIFIED ACCORDING
              C       TO STRUCTURE OF RAMS DATA AND TREATMENT OF MISSING DATA.
ISN 0045            PARM(1)=UBAR
ISN 0046            I=0
ISN 0047        450 I=I+1
ISN 0048            IF (RAMS(2,I).GT.360.) GO TO 450
ISN 0050            PARM(2)=RAMS(2,I)
ISN 0051            I=0
ISN 0052        460 I=I+1
ISN 0053            IF ((RAMS(3,I).GE.99.).OR.(RAMS(4,I).GE.99.)) GO TO 460
ISN 0055            PARM(3)=RAMS(3,I)
ISN 0056            PARM(8)=RAMS(4,I)
ISN 0057            PARM(4)=S
ISN 0058            PARM(9)=RAMS(6,1)
              C
ISN 0059            IF((KWIND.EQ.3).OR.(KWIND.EQ.4)) CALL WINDGR  (U1,V1,UU,VV
                  & ,NEAR,USTN,VSTN,XSUTM,YSUTM,IUTM,JUTM,DX,DY,IM,JM,IN,JN,NS)
              C
              C *** COMPUTE UPPER LAYER U AND V ***
              C
ISN 0061            LX=1
ISN 0062            CALL      UVZF      (U,V,U1,V1,UZF,VZF,WZF,Z,IM,JM,KM,KN,LX)
              C
              C *** COMPUTE VERTICAL WIND COMPONENT ***
              C
ISN 0063            IF (LWW.GT.0)     CALL      WWZF
```

```
                  & (U,V,W,DX,DY,DZ,IM,JM,KM,KN,LWW,JUNIT)
  ISN 0065          GO TO 700
              C
  ISN 0066        500 WRITE (JUNIT ,870 )
  ISN 0067          GO TO 700
  ISN 0068        600 WRITE (JUNIT ,880 )
              C
  ISN 0069        700 CONTINUE
              C
  ISN 0070          RETURN
              C     DEBUG SUBCHK
  ISN 0071          END
*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*      SOURCE STATEMENTS =      70 ,PROGRAM SIZE =      2954

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                   109K BYTES OF CORE NOT USED
```

LEVEL 21.6 (MAY 72)   OS/360 FORTRAN H

```
      COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                         SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE WRITES (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IUTM,JUTM
                     *                 ,IFORM,JUNIT,RATIO,TITLE)
                C
                C.... THIS ROUTINE WRITES Q(ISIZE,JSIZE) ON UNIT=JUNIT IN FORM=IFORM.
                C     IFORM = NO. OF COLUMES TO BE PRINTED ON ONE LINE.
                C     II    = DUMMY ARRAY FOR TEMPORAL STORAGE
                C
  ISN 0003            DIMENSION RFMT1(10),RFMT2(10),RFMT3(10),TITLE(10),FMTDS(10)
                     *          ,Q(IXMAX,IYMAX),IUTM(ISIZE),JUTM(JSIZE),II(130)
                C
  ISN 0004            LOGICAL*1 FMT1(40),FMT2(40),FMT3(40),FM(11)
                C
  ISN 0005            EQUIVALENCE (FM(1),IFM1),(FM(5),IFM2)
                     * ,(FMT1(1),RFMT1(1)),(FMT2(1),RFMT2(1)),(FMT3(1),RFMT3(1))
                C
  ISN 0006            DATA RFMT1/'(5X,','5HXU'.'TM=,'.'30I4',')   )',5*'    '/
                     *    ,RFMT2/'(5X,','5H  '.' I=,'.'30I4',',/ )',5*'    '/
                     *    ,RFMT3/'(1X,','2I4,','1X ,','30I4',')   )',5*'    '/
                     *    ,FM/'1','2','3','4','5','6','7','8','9','0',' '/
                C
  ISN 0007            RETURN
                C
                C     ------------------------------------------------
                C
                C *** PRINT HORIZONTAL SURFACE OF VARIABLE Q ***
                C
  ISN 0008            ENTRY WRITEX (Q,IXMAX,IYMAX,ISIZE,JSIZE,IFORM,JUNIT,RATIO,TITLE)
                C
  ISN 0009       8000 FORMAT (10A4,F7.2)
  ISN 0010       8040 FORMAT ('    ..... PAGE = ',I5/)
                C8040 FORMAT('1   ..... PAGE    ',I5,/5X,'IUTM=',30I4,/,5X,'    I=',40I3//)
                C8050 FORMAT (1X,I4,I4,1X,30I4)
                C
                C.... DETERMINE DOMAIN OF ARRAY Q TO BE PRINTED.
  ISN 0011            JEND=JBEG+JSIZE-1
  ISN 0012            IEND=IBEG+ISIZE-1
                C
                C.... FIX VARIABLE FORMAT STATEMENTS
  ISN 0013            IF (IFORM .EQ. 0) GO TO 500
  ISN 0015            IF1=IFORM/10
  ISN 0016            IF (IF1.GT.9) IF1=9
  ISN 0018            IF2=IFORM-IF1*10
  ISN 0019            IF (IF2.EQ.0) IF2=10
  ISN 0021            IF (IF1.EQ.0) IF1=11
  ISN 0023            IF3=120/IFORM
  ISN 0024            IF (IF3.GT.9) IF3=9
  ISN 0026            FMT1(13)=FM(IF1)
  ISN 0027            FMT2(13)=FM(IF1)
  ISN 0028            FMT3(13)=FM(IF1)
  ISN 0029            FMT1(14)=FM(IF2)
  ISN 0030            FMT2(14)=FM(IF2)
  ISN 0031            FMT3(14)=FM(IF2)
  ISN 0032            FMT1(16)=FM(IF3)
  ISN 0033            FMT2(16)=FM(IF3)
  ISN 0034            FMT3(16)=FM(IF3)
```

```
              C
              C.... DETERMINE NO. OF PAGES PER ARRAY Q TO BE PRINTED
ISN 0035            IPT=(ISIZE-1)/IFORM+1
              C
              C.... FIX RATIO OF PRINTED VALUE TO ACTUAL VALUE OF Q
ISN 0036            IF (RATIO.NE.0.0) GO TO 100
ISN 0038            RATIO=1000.0
ISN 0039            IF (IFORM .GE. 30) RATIO=100.
ISN 0041        100 CONTINUE
              C
              C.... PRINT THE SPECIFIED TITLE
ISN 0042            WRITE (JUNIT,8000) TITLE,RATIO
              C
              C *** ENTERING PRINTING LOOP ***
ISN 0043            DO 400 IP=1,IPT
ISN 0044            IA=(IP-1)*IFORM+IBEG
ISN 0045            IB=IA+IFORM-1
              C
              C.... FIX INDICES
ISN 0046            DO 200 I=IA,IB
ISN 0047            I1=I-IA+1
ISN 0048            II(I1)=I
ISN 0049            IF ((II(I1).GT.100).AND.(IFORM.GT.30)) II(I1)=II(I1)-100
ISN 0051        200 CONTINUE
              C
              C.... PRINT PAGE NO., INDICES & UTM COORDINATES
ISN 0052            IAX=IA-IBEG+1
ISN 0053            IBX=IAX+IFORM-1
              C     WRITE (JUNIT,8040) IP,(IUTM(I),I=IAX,IBX),(II(I1),I1=IA,IB)
ISN 0054            WRITE (JUNIT,8040) IP
ISN 0055            WRITE (JUNIT,FMT1) (IUTM(I),I=IAX,IBX)
ISN 0056            WRITE (JUNIT,FMT2) (II(I1),I1=IA,IB)
              C
              C.... PRINT ARRAY Q
ISN 0057            DO 320 JJ=1,JSIZE
ISN 0058            J=(JEND+1)-JJ
ISN 0059            JUTMJ=JUTM(JSIZE+1-JJ)
ISN 0060            DO 300 I=IA,IB
ISN 0061            I1=I-IA+1
ISN 0062        300 II(I1)=Q(I,J)*RATIO
              C     WRITE (JUNIT,8050) JUTMJ,J,(II(I1),I1=IA,IB)
ISN 0063            WRITE (JUNIT,FMT3) JUTMJ,J,(II(I1),I1=IA,IB)
ISN 0064        320 CONTINUE
ISN 0065        400 CONTINUE
              C
ISN 0066        500 CONTINUE
              C
ISN 0067            RETURN
              C
              C     ----------------------------------------------------
              C....
              C *** PRINT VERTICAL CROSS SECTION OF VARIABLE A FOR I=IMC AND J=JMC.
              C
ISN 0068            ENTRY WRITEZ (A,Z,RATIO,IM,JM,KM,IMC,JMC,TITLA)
              C....
ISN 0069            DIMENSION A(IM,JM,KM),Z(KM)
              C
ISN 0070       8500 FORMAT (/' ** VERTICAL PROFILE OF ',A4,' AT I=',I3,'; J=',I3,'**')
```

```
ISN 0071      8502 FORMAT (/5X,'XUTM=',30I4)
ISN 0072      8504 FORMAT (5X,'    I=',30I4/4X,'Z',3X,'K')
ISN 0073      8506 FORMAT (/5X,'YUTM=',30I4)
ISN 0074      8508 FORMAT (5X,'    J=',30I4/4X,'Z',3X,'K')
ISN 0075      8510 FORMAT (1X,2I4,1X,30I4)
             C
ISN 0076           WRITE (JUNIT,8500) TITLA,IMC,JMC
             C
             C.... PRINT Y-Z CROSS SECTION
ISN 0077           IA=1
ISN 0078           IB=IM
ISN 0079           IF (IM.LE.30) GO TO 605
ISN 0081           IA=(IM-30)/2+1
ISN 0082           IB=IA-1+30
ISN 0083       605 CONTINUE
ISN 0084           WRITE (JUNIT,8502) (IUTM(I),I=IA,IB)
ISN 0085           WRITE (JUNIT,8504) (I,I=IA,IB)
ISN 0086           J=JMC
ISN 0087           DO 620 KK=1,KM
ISN 0088           K=KM+1-KK
ISN 0089           KZ=Z(K)
ISN 0090           DO 610 I=IA,IB
ISN 0091       610 II(I)=A(I,J,K)*RATIO
ISN 0092           WRITE (JUNIT,8510) KZ,K,(II(I),I=IA,IB)
ISN 0093       620 CONTINUE
             C
             C.... PRINT X-Z CROSS SECTION
ISN 0094           JA=1
ISN 0095           JB=JM
ISN 0096           IF (JM.LE.30) GO TO 625
ISN 0098           JA=(JM-30)/2+1
ISN 0099           JB=JA-1+30
ISN 0100       625 CONTINUE
ISN 0101           WRITE (JUNIT,8506) (JUTM(J),J=JA,JB)
ISN 0102           WRITE (JUNIT,8508) (J,J=JA,JB)
ISN 0103           I=IMC
ISN 0104           DO 640 KK=1,KM
ISN 0105           K=KM+1-KK
ISN 0106           KZ=Z(K)
ISN 0107           DO 630 J=JA,JB
ISN 0108       630 II(J)=A(I,J,K)*RATIO
ISN 0109           WRITE (JUNIT,8510) KZ,K,(II(J),J=JA,JB)
ISN 0110       640 CONTINUE
             C
ISN 0111           RETURN
             C     DEBUG SUBCHK
ISN 0112           END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      111 ,PROGRAM SIZE =      4028

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                97K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H

        COMPILER OPTIONS   NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE WWFLUX ( CP1, C,W,WZF,Z,DX,DY,DZ,EK,FK,
                      *                       RDX,RDY,RDZ,IM,JM,KM,IMJM,IJKM,IJKN)
                  C
                  C.... THIS SUBROUTINE COMPUTES VERTICAL ADVECTION TERM OF
                  C....  CONCENTRATION EQUATION.
                  C.... SAME NUMERICAL SCHEME AS USED IN SUBROUTINE UVFLUX IS APPLIED
                  C.... IN THIS SUBROUTINE.
                  C
  ISN 0003              DIMENSION
                      *        CP1(IJKM),C(IJKM),W(IJKN),WZF(KM)
                      *        ,DX(IM),DY(JM),DZ(KM),RDX(IM),RDY(JM),RDZ(KM),Z(KM)
                      *        ,EK(KM),FK(KM)
                  C
  ISN 0004              COMMON    /AADATA/
                      *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                      *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                      *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                      *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                      *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                      *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                  C
                  C *** COEFFICIENTS OF FINITE DIFF. FORMULA ***
  ISN 0005              LX=1
  ISN 0006              LY=IM
  ISN 0007              LZ=IMJM
  ISN 0008              KN=IJKN/IMJM
  ISN 0009              KKVMAX=(KN-1)*IMJM
  ISN 0010              DO 20  K= 2, KM
  ISN 0011              FK(K-1)= 1.0/(1.0+RDZ(K))
  ISN 0012           20 EK(K  )= RDZ(K)* FK(K-1)
  ISN 0013              EK(1)= 0.5
  ISN 0014              FK(KM)=0.5
                  C
                  C *** PROCESS INTERIOR GRID POINT ***
  ISN 0015              KK=0
  ISN 0016              DO 300 K=2,KM1
  ISN 0017              ZK=Z(K)
  ISN 0018              KK=KK+ IMJM
  ISN 0019              KKV=KK
                  C     SAME W FOR K GT KN .....
  ISN 0020              IF ( K .GT. KN) KKV=KKVMAX
  ISN 0022              DTDZ=     DT/DZ(K)
  ISN 0023              JJ=0
  ISN 0024              DO 200 J= 2,JM1
  ISN 0025              JJ=JJ+ IM
  ISN 0026              DO 200 I= 2,IM1
  ISN 0027              IJ= I+JJ
  ISN 0028              IJK = IJ+KK
  ISN 0029              IJKV=IJ+KKV
  ISN 0030              AF1= DTDZ*W(IJKV)*WZF(K)
  ISN 0031              A3= AF1*AF1
  ISN 0032              CIJK= C(IJK)
  ISN 0033              CIJK1= C(IJK+LZ)
  ISN 0034              CPIJK =CP1(IJK)
  ISN 0035              CPIJK1=CP1(IJK+LZ)
```

```
   ISN 0036            T1= CIJK+CIJK1
   ISN 0037            T3= CIJK-CIJK1
   ISN 0038            WF= AF1*T1+ A3*T3
   ISN 0039            CCIJK=CPIJK-EK(K)*WF
   ISN 0040            CCIJK1= CPIJK1+FK(K)*WF
                 C     CHECK FLUX SO THAT FLUX OUT IS LESS THAN MASS IN A CELL ....
   ISN 0041            IF ((CCIJK .GE. 0.0) .AND. (CCIJK1 .GE. 0.0)) GO TO 110
   ISN 0043            IF ( CCIJK .LT. 0.) WF=CPIJK/EK(K)
   ISN 0045            IF ( CCIJK1 .LT. 0.0) WF=-CPIJK1/FK(K)
   ISN 0047            CCIJK=CPIJK-EK(K)*WF
   ISN 0048            CCIJK1= CPIJK1+FK(K)*WF
   ISN 0049        110 CP1(IJK)=CCIJK
   ISN 0050            CP1(IJK+LZ)=CCIJK1
   ISN 0051        200 CONTINUE
   ISN 0052        300 CONTINUE
                 C
                 C *** PROCESS UPPER BOUNDARY GRID POINTS ***
                 C.... LWTOP IS A CONTROL PARAMETER FOR CONDITON AT TOP BOUNDARY.
                 C.... IF LWTOP=0, VERTICAL ADVECTION IS ASSUMED EQUAL TO ZERO AT
                 C      BOUNDARY
                 C
   ISN 0053            IF ( LWTOP .EQ. 0) RETURN
                 C
   ISN 0055            KK=KM1*IMJM
   ISN 0056            JJ=-IM
   ISN 0057            DO 500 J=1,JM
   ISN 0058            JJ=JJ+ IM
   ISN 0059            DO 400 I=1,IM
   ISN 0060            IJ= I+JJ
   ISN 0061            IJK = IJ+KK
   ISN 0062            IJKV=IJ+KKVMAX
   ISN 0063            AF1= DTDZ*W(IJKV)*WZF(K)
   ISN 0064            IF ( AF1 .LT. 0.0) GO TO 400
   ISN 0066            CIJK= C(IJK)
   ISN 0067            CIJK1= C(IJK-LZ)
   ISN 0068            T3= CIJK-CIJK1
   ISN 0069            WF=AF1*T3
   ISN 0070            IF ( CIJK1 .LT. WF) WF= CIJK1
   ISN 0072            CP1(IJK)= CIJK+WF
   ISN 0073        400 CONTINUE
   ISN 0074        500 CONTINUE
                 C
   ISN 0075            RETURN
                 C     DEBUG SUBCHK
   ISN 0076            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      75 ,PROGRAM SIZE =      1738

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                          113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                     C
   ISN 0002                SUBROUTINE WWZF (U,V,W,DX,DY,DZ,IM,JM,KM,KN,LWW,JUNIT)
                     C
                     C.... IT CALCULATES VERTICAL WIND W FROM CONTINUITY EQ.
                     C         & SMOOTHES W.
                     C
   ISN 0003                DIMENSION U(IM,JM,KN),V(IM,JM,KN),W(IM,JM,KN)
   ISN 0004                DIMENSION DX(IM),DY(JM),DZ(KM)
                     C
   ISN 0005                IF (LWW.EQ.0) RETURN
                     C
   ISN 0007                DXM= 0.5/DX(1)
   ISN 0008                DYM= 0.5/DY(1)
   ISN 0009                IM1=IM-1
   ISN 0010                JM1=JM-1
                     C
                     C     LWW=0,NO VERTICAL WIND W;=1,W FROM CONTINUETY EQ;
                     C     =2, W ARE SMOOTHED AGAIN BY 9 POINTS AVERAGE
                     C
   ISN 0011                DO 100 K=2,KN
   ISN 0012                DZM= 0.5*(DZ(K)+ DZ(K-1))
   ISN 0013                DZDY=DZM*DYM
   ISN 0014                DZDX=DZM*DXM
   ISN 0015                DO 100 J=2,JM1
   ISN 0016                DO 100 I=2,IM1
   ISN 0017                DUDX= DZDX*(U(I-1,J,K-1)-U(I+1,J,K-1))
   ISN 0018                DVDY= DZDY*(V(I,J-1,K-1)-V(I,J+1,K-1))
   ISN 0019                W(I,J,K)= W(I,J,K-1)+ DUDX+DVDY
   ISN 0020            100 CONTINUE
                     C
   ISN 0021                IF (LWW .LT. 2) GO TO 300
   ISN 0023                DO 220 L=1,2
   ISN 0024                DO 220 K=2,KN
   ISN 0025                DO 200 J=1,JM
   ISN 0026                DO 200 I=1,IM
   ISN 0027            200 W(I,J,1)= W(I,J,K)
                     C
   ISN 0028                DO 210 J=2,JM1
   ISN 0029                DO 210 I=2,IM1
   ISN 0030                WTOT= W(I-1,J,1)+W(I,J,1)+W(I+1,J,1)+W(I-1,J-1,1)+W(I,J-1,1)
                         * +W(I+1,J-1,1)+W(I-1,J+1,1)+W(I,J+1,1)+W(I+1,J+1,1)
   ISN 0031            210 W(I,J,K)=0.111111*WTOT
   ISN 0032            220 CONTINUE
                     C
   ISN 0033                DO 230 J=1,JM
   ISN 0034                DO 230 I=1,IM
   ISN 0035            230 W(I,J,1)=0.0
   ISN 0036            300 CONTINUE
                     C
   ISN 0037                RETURN
                     C     DEBUG SUBCHK
   ISN 0038                END

*OPTIONS IN EFFECT*        NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*        SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
```

```
*STATISTICS*     SOURCE STATEMENTS =      37 ,PROGRAM SIZE       1802
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                    109K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                   OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
  ISN 0002              SUBROUTINE XYDIFF ( CP1,  C,AKH,AKZ,AKF,RDX,RDY,DXS,DYS,
                       *                   IM,JM,KM,IMJM,IJKM)
                  C.... THIS SUBROUTINE COMPUTES THE HORIZONTAL DIFFUSION TERMS IN
                  C....  CONCENTRATION EQUATION.
                  C.... THE SECOND ORDER CENTRAL FINITE DIFFERENCE SCHEME IS USED.
                  C.... THE 3-D VARIABLE A(I,J,K) IS REPRESENTED BY VECTOR A(IJK).
                  C.... AKH IS HORIZONTAL EDDY EXCHANGE COEFFICIENT.
                  C
  ISN 0003              DIMENSION
                       *       CP1(IJKM),C(IJKM),AKH(KM),AKZ(IMJM),AKF(KM)
                       *      ,RDX(IM),DXS(IM),RDY(JM),DYS(JM),DTKXSI(40),DTKYSI(40)
                  C
                  C     DATA RDXA/1.0/,RDYA/1.0/,RDXB/2.0/,RDYB/2.0/
                  C
  ISN 0004              COMMON     /AADATA/
                       *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                       *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                       *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                       *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                       *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                       *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,UO,PHIFHZ,HFZ
                  C
  ISN 0005              LX=1
  ISN 0006              LY=IM
  ISN 0007              KK=0
                  C *** COMPUTATION ***
                  C
  ISN 0008              DO 300 K= 2,KM
  ISN 0009              DTKH= DT*AKH(K)*AKF(K)
  ISN 0010              DO 10 J=1,JM
  ISN 0011           10 DTKYSI(J)=DTKH/DYS(J)
  ISN 0012              DO 20 I=1,IM
  ISN 0013           20 DTKXSI(I)=DTKH/DXS(I)
                  C     DTKXS= DTKH/DXS(1)
                  C     DTKYS= DTKH/DYS(1)
  ISN 0014              KK=KK+ IMJM
                  C
                  C.... INTERIOR REGION, STRAIGHT FURWARD CALCULATION
                  C *** PROCESS INTERIOR GRID POINTS ***
                  C
  ISN 0015              JJ=0
  ISN 0016              DO 200 J= 2,JM1
  ISN 0017              JJ=JJ+ IM
  ISN 0018              RDYB=1.0+RDY(J)
  ISN 0019              DO 100 I= 2,IM1
  ISN 0020              IJ= I+JJ
  ISN 0021              IJK = IJ+KK
  ISN 0022              RDXB=1.0+RDX(I)
  ISN 0023              DXC= C(IJK+LX)-RDXB*C(IJK)+RDX(I)*C(IJK-LX)
  ISN 0024              DYC= C(IJK+LY)-RDYB*C(IJK)+RDY(J)*C(IJK-LY)
  ISN 0025              CP1(IJK)= CP1(IJK)+(DTKYSI(J)*DYC+DTKXSI(I)*DXC)*AKZ(IJ)
  ISN 0026          100 CONTINUE
                  C
                  C     EAST-WEST BOUNDRIES,  DIFFUS IN Y ONLY,  DIFF IS ZERO IN X
                  C *** PROCESS EAST AND WEST BOUNDARY GRID POINTS ***
```

```
                 C
   ISN 0027            DO 150 I= 1, IM, IM1
   ISN 0028            IJ= I+JJ
   ISN 0029            IJK = IJ+KK
   ISN 0030            DYC= C(IJK+LY)-RDYB*C(IJK)+RDY(J)*C(IJK-LY)
   ISN 0031            CP1(IJK)= CP1(IJK)+DTKYSI(J)*DYC*AKZ(IJ)
   ISN 0032        150 CONTINUE
   ISN 0033        200 CONTINUE
                 C
                 C     NORTH-SOUTH BOUNDRY, DIFFUS IN X ONLY, DIFF IS ZERO IN Y
                 C *** PROCESS SOUTH AND NORTH BOUNDARY GRID POINTS ***
                 C
   ISN 0034            DO 220 J= 1, JM, JM1
   ISN 0035            JLY=(J-1)*LY
   ISN 0036            DO 210 I= 2, IM1
   ISN 0037            ILX= (I-1)*LX
   ISN 0038            IJ= 1+ ILX+JLY
   ISN 0039            IJK = IJ+KK
   ISN 0040            DXC= C(IJK+LX)-(1.0+RDX(I))*C(IJK)+RDX(I)*C(IJK-LX)
   ISN 0041            CP1(IJK)= CP1(IJK)+DTKXSI(I)*DXC*AKZ(IJ)
   ISN 0042        210 CONTINUE
   ISN 0043        220 CONTINUE
   ISN 0044        300 CONTINUE
                 C
   ISN 0045            RETURN
                 C     DEBUG SUBCHK
   ISN 0046            END

*OPTIONS IN EFFECT*     NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*     SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*    SOURCE STATEMENTS =      45 ,PROGRAM SIZE =      1798

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                      113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

        COMPILER OPTIONS    NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE XYUTMS (IUTM,JUTM,IXBEG,IYBEG,IBEG,JBEG,DX,DY
                     &                 ,DXA,DYA,IM,JM)
                C
                C.... THIS ROUTINE COMPUTES UTM COORDINATES FOR ALL NUMERICAL GRIDS.
                C.... ENTRY XYUTMI CONVERT (X,Y) FROM UTM COORDINATES TO NUMERICAL GRID.
                C
  ISN 0003            DIMENSION IUTM(IM),JUTM(JM),DX(IM),DY(JM)
                C
  ISN 0004            IM1=IM-1
  ISN 0005            JM1=JM-1
  ISN 0006            DXAI=1.0/DXA
  ISN 0007            DYAI=1.0/DYA
  ISN 0008            IUTM(1)=IXBEG+IBEG-1
  ISN 0009            JUTM(1)=IYBEG+JBEG-1
  ISN 0010            DO 20 J=1,JM1
  ISN 0011            JINTVL=DY(J)*DYAI+0.2
  ISN 0012            JUTM(J+1)=JUTM(J)+JINTVL
  ISN 0013         20 CONTINUE
                C
  ISN 0014            DO 40 I=1,IM1
  ISN 0015            IINTVL=DX(I)*DXAI+0.2
  ISN 0016            IUTM(I+1)=IUTM(I)+IINTVL
  ISN 0017         40 CONTINUE
  ISN 0018            IUTMAX=IUTM(IM)+DX(IM)*DXAI+0.2
  ISN 0019            JUTMAX=JUTM(JM)+DY(JM)*DYAI+0.2
                C
  ISN 0020            RETURN
                C
                C     ------------------------------------------------------------
                C
  ISN 0021            ENTRY XYUTMI       (XUTM,YUTM,IDIMAX,XD,YD,IDIM,LWAY)
                C
                C.... CHANGE (XUTM,YUTM) FROM UTM TO GRID UNIT (XD,YD) WITH ORIGIN(0,0).
                C       THE (0,0) IS AT (IUTM(1),JUTM(1)).
                C
  ISN 0022            DIMENSION XUTM(IDIMAX),YUTM(IDIMAX),XD(IDIM),YD(IDIM)
                C
  ISN 0023            DO 200 K=1,IDIM
  ISN 0024            XI=XUTM(K)
  ISN 0025            YJ=YUTM(K)
  ISN 0026            LXI=XI
  ISN 0027            LYJ=YJ
  ISN 0028            IF ((LYJ.LT.JUTM(1)).OR.(LYJ.GE.JUTMAX)) GO TO 130
  ISN 0030            DO 120 J=2,JM
  ISN 0031            IF (LYJ.GE.JUTM(J)) GO TO 120
                C
                C.... FIND THE JUTM(J) GREATER THAN LYJ.
  ISN 0033            YJJ=J-2
  ISN 0034            DYB=(YJ-JUTM(J-1))/(DY(J-1)*DYAI)
  ISN 0035            YJ=YJJ+DYB
  ISN 0036            GO TO 140
  ISN 0037        120 CONTINUE
                C
                C.... YI IS OUTSIDE THE REGION.
  ISN 0038        130 IF (LYJ.LT.JUTM(1)) YJ=(YJ-JUTM(1))/(DY(1)*DYAI)
```

```
   ISN 0040          IF (LYJ.GE.JUTMAX)  YJ=(YJ-JUTMAX)/(DY(JM)*DYAI)+JM
   ISN 0042      140 CONTINUE
   ISN 0043          IF ((LXI.LT.IUTM(1)).OR.(LXI.GE.IUTMAX)) GO TO 160
   ISN 0045          DO 150 I=1,IM
   ISN 0046          IF (LXI.GE.IUTM(I)) GO TO 150
               C
               C.... FIND THE IUTM(I) GREATER THAN LXI.
   ISN 0048          XII=I-2
   ISN 0049          DXB=(XI-IUTM(I-1))/(DX(I-1)*DXAI)
   ISN 0050          XI=XII+DXB
   ISN 0051          GO TO 180
   ISN 0052      150 CONTINUE
               C
               C.... XI IS OUTSIDE THE REGION.
   ISN 0053      160 IF (LXI.LT.IUTM(1)) XI=(XI-IUTM(1))/(DX(1)*DXAI)
   ISN 0055          IF (LXI.GE.IUTMAX)  XI=(XI-IUTMAX)/(DX(IM)*DXAI)+IM
   ISN 0057      180 CONTINUE
   ISN 0058          XD(K)=XI
   ISN 0059          YD(K)=YJ
   ISN 0060      200 CONTINUE
               C
   ISN 0061          RETURN
               C     DEBUG SUBCHK
   ISN 0062          END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*    SOURCE STATEMENTS =       61 ,PROGRAM SIZE =     2316

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                             113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                              OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                          SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                  C
 ISN 0002               SUBROUTINE ZZDIFF ( CP1,  C,AKZ,AKF,RDZ,DZS,ZM,EK,FK,
                      *                     IM,JM,KM,IMJM,IJKM)
                  C
                  C     IT CALCULATES THE VERTICAL DIFFUSION BY IMPLICIT SECOND CENTRAL
                  C     SCHEME,FOR VARIABLE GRIDS AND VARIABLE EDDY DIFUSIVITY.
                  C     CP1=NEW , C=OLD,AKZ=EDDY DIFUSIIVITY, RDZ=GRID RATIO,
                  C     DZS=GRID SQUARE, EK,FK=TEMPORARY STORAGE, ZM=HEIGHT...
                  C
 ISN 0003               DIMENSION
                      *          CP1(IJKM),C(IJKM),AKZ(IMJM)
                      *        ,RDZ(KM),DZS(KM),ZM(KM),EK(KM),FK(KM),AKF(KM)
                      *        ,AKKK(20),CKKK(20)
                  C
 ISN 0004               COMMON    /AADATA/
                      *   IM1,JM1,KM1,JUNIT,KUNITC,KUNITG,KUNITP,KUNITS,KUNITW
                      *  ,IYR,IMO,IDAY,IHR,ITM,ITMHR,ITSEC,ITOTHR,ITSTEP,DT,TM,TSEC
                      *  ,LPRINT,LTSTOP,LTSOUS,LTWIND
                      *  ,LWRITE(10),LSOUS(2),LTOP,LWTOP,LPQ,LWW,LWIND,KWIND,LCRUN,LCHEM
                      *  ,RAMS(6,25),PARM(10),A1(4),AK,HG,HP,HS,OLMIN,DCMIN
                      *  ,PMAX,PMIN,RIB,ZMAX,ZRPQ,ZRISE,QBTOT,PQBTOT,U0,PHIFHZ,HFZ
                  C
                  C     AKZF IS TO SPECIFY THE VERTICAL SHAPE FUNCTION OF EDDY DIFUSIVITY
                  C     STABILITY CRITERIA IS (DT*AKZ/DZS) .LT. 1/(4*(1-RATIO)) TO AVOID
                  C     THE OSCILLATORY MODES.
                  C     LTOP IS TO SPECIFY THE TOP BOUNDRY CONDITIONS,=0, THEN C=0;
                  C     =1,THEN REFLECTED BOUNDRY.....
                  C
                  C.... RATIO=0.0,TOTALLY EXPLICIT; =1,TOTALLY IMPLICIT....
 ISN 0005               RATIO=0.98
 ISN 0006               RATIOA=(1.0/RATIO-1.0)
 ISN 0007               RATIO2=2.0*RATIO
 ISN 0008               DTT=DT
 ISN 0009               DO 10 K=1,KM1
 ISN 0010               AKZFK= AKF(K)
 ISN 0011               AKKK(K)= DTT*AKZFK/DZS(K)
 ISN 0012            10 CKKK(K+1) =DTT*AKZFK/DZS(K+1)*RDZ(K+1)
 ISN 0013               AKKK(KM)=0.0
 ISN 0014               CKKK(1)=0.0
                  C
 ISN 0015               LZ=IMJM
 ISN 0016               JJ=-IM
 ISN 0017               DO 300 J=1,JM
 ISN 0018               JJ=JJ+IM
 ISN 0019               DO 200 I=1,IM
 ISN 0020               IJ=I+JJ
 ISN 0021               AKIJ= AKZ(IJ)*RATIO
 ISN 0022               AKIJ2=2.0*AKIJ
 ISN 0023               AK1= AKIJ2*AKKK(1)
 ISN 0024               BK1= 1.0+AK1
 ISN 0025               CKKM= AKIJ2*CKKK(KM)
                  C     CK1= 0.0
 ISN 0026               BKKM= 1.0+CKKM
                  C     AKKM= 0.0
 ISN 0027               DK1= CP1(IJ)+AK1*(C(IJ+LZ)-C(IJ))*RATIOA
 ISN 0028               KMIJ= KM1*IMJM+IJ
```

```
ISN 0029          DKKM= CP1(KMIJ)+CKKM*(C(KMIJ-LZ)-C(KMIJ))*RATIOA
ISN 0030          EK(1)= AK1/BK1
ISN 0031          FK(1)= DK1/BK1
ISN 0032          KK=0
ISN 0033          DO 100 K=2,KM1
ISN 0034          KK=KK+IMJM
ISN 0035          IJK= KK+IJ
ISN 0036          AKK= AKIJ*AKKK(K)
ISN 0037          CKK= AKIJ*CKKK(K)
ISN 0038          BKK= 1.0+AKK+CKK
ISN 0039          DKG=          AKK*C(IJK+LZ)-(AKK+CKK)*C(IJK)+CKK*C(IJK-LZ)
ISN 0040          DKK=CP1(IJK)+DKG*RATIOA
ISN 0041          TEMP=1.0/(BKK-CKK*EK(K-1))
ISN 0042          EK(K)= AKK*TEMP
ISN 0043          FK(K)= (DKK+CKK*FK(K-1))*TEMP
ISN 0044      100 CONTINUE
            C
ISN 0045          EK(KM)=0.0
ISN 0046          FK(KM)= (DKKM+CKKM*FK(KM1))/(BKKM-CKKM*EK(KM1))
ISN 0047          CP1(KMIJ)=FK(KM)
ISN 0048          IF (LTOP .EQ. 0) CP1(KMIJ)= 0.0
ISN 0050          KK= KM1*IMJM
ISN 0051          DO 150 L= 1, KM1
ISN 0052          KK= KK-IMJM
ISN 0053          IJK= IJ+KK
ISN 0054          K= KM-L
ISN 0055          CP1(IJK)= FK(K)+EK(K)*CP1(IJK+LZ)
ISN 0056      150 CONTINUE
ISN 0057      200 CONTINUE
ISN 0058      300 CONTINUE
            C
ISN 0059          RETURN
            C     DEBUG SUBCHK
ISN 0060          END
```

```
*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       59 ,PROGRAM SIZE =      1744

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                            113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                               OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF
                C
  ISN 0002            SUBROUTINE     ZZGRID     (ZA,ZB,Z,IMA,IMB,KM)
                C
                C.... IT CONVERTS ZA IN METER UNIT INTO GRID UNIT & STORES IN ZB.
                C
  ISN 0003            DIMENSION ZA(IMA),ZB(IMB),Z(KM)
                C
  ISN 0004            IMAX=IMA
  ISN 0005            IF(IMAX .LT. IMB) IMAX=IMB
  ISN 0007            DO 200 I=1,IMAX
  ISN 0008            IF(ZA(I) .LE. Z(1)) GO TO 110
  ISN 0010            IF(ZA(I) .GE.Z(KM)) GO TO 120
  ISN 0012            DO 100 K=1,KM
  ISN 0013            IF (ZA(I) .GT. Z(K)) GO TO 100
  ISN 0015            K1=K-1
  ISN 0016            ZB(I)=K1+(ZA(I)-Z(K1))/(Z(K)-Z(K1)) - 1
  ISN 0017            GO TO 200
  ISN 0018        100 CONTINUE
  ISN 0019        110 ZB(I)=0.0
  ISN 0020            GO TO 200
  ISN 0021        120 ZB(I)=KM-1+(ZA(I)-Z(KM))/(Z(KM)-Z(KM-1))
  ISN 0022        200 CONTINUE
                C
  ISN 0023            RETURN
                C     DEBUG SUBCHK
  ISN 0024            END

*OPTIONS IN EFFECT*     NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*     SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID,NOXREF

*STATISTICS*    SOURCE STATEMENTS       23 ,PROGRAM SIZE       700

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                              125K BYTES OF CORE NOT USED

*STATISTICS*    1 DIAGNOSTICS THIS STEP, HIGHEST SEVERITY CODE IS  4
```

```
F128-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED LET,LIST,MAP
         DEFAULT OPTION(S) USED -  SIZE=(225280,57344)
IEW0000      INCLUDE SYSLIB(WERCOMX)
IEW0000        ENTRY MAIN
```

MODULE MAP

```
 CONTROL SECTION                             ENTRY

   NAME     ORIGIN  LENGTH                     NAME     LOCATION     NAME     LOCATION     NAME     LOCATION     NAME     LOCATION

 WERCOM        00    1238
                                             EXIT          54C     DVCHK         5DE     OVERFL        608     WERTIME       768
                                             WERPARMS      9A8
 IHOECOMH    1238     DE0
                                             IBCOM#       1278     IB081971     1278     FDIOCS#      1334     INTSWTCH     2000
 IHOCOMH2    2018     91D
                                             SEQDASD      232A
 IHOEFIOS    2938    1114
                                             FIOCS#       2938     FIOCSBEP     293E
 IHOFIOS2    3A50     5CC
 IHOERRM     4020     624
                                             ERRMON       4020     IHOERRE      4038
 IHOETRCH    4648     2A6
                                             IHOTRCH      4648     ERRTRA       4650
 IHOFCONI    48F0     2FD
                                             FQCONI#      48F0
 IHOFCONO    48F0     4C2
                                             FQCONO#      48F0
 IHOFCVTH    50B8     A07
                                             ADCON#       50B8     FCVAOUTP     5162     FCVLOUTP     51F2     FCVZOUTP     534E
                                             FCVIOUTP     56F6     FCVEOUTP     57E8     FCVCOUTP     57E8     INT6SWCH     5A40
 IHOFTEN     5AC0     198
                                             FTEN#        5AC0
 IHOUATBL    5C58     208
 IHOUOPT     5E60     338
 MAIN        6198    CED4
 AACOMP     13070     C5A
 AKZCAL     13CD0     686
 CADJUS     14358     3F8
 CCHECK     14750     17C
 CHEMIC     148D0     218
 CONSIN     14AE8     434
                                             CDTOTP      14EBC
 DIMENS     14F20     870
                                             DIMEN1      15648     HHCALC      15724
 DTTEST     15790     74A
 GEOIN      15EE0     B76
 OUTAPE     16A58     70E
 POSITV     17168     198
 PRINTS     17300    248C
                                             PRINTA      1967C     PRINTB      196C4     PRINTC      19718
 SHIFTN     19790     32A
```

```
F128-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED LET,LIST,MAP
          DEFAULT OPTION(S) USED -  SIZE=(225280,57344)
IEW0000      INCLUDE SYSLIB(WERCOMX)
IEW0000        ENTRY MAIN

                                                              MODULE MAP

  CONTROL SECTION                        ENTRY

    NAME    ORIGIN  LENGTH                 NAME    LOCATION     NAME    LOCATION     NAME    LOCATION     NAME    LOCATION

  WERCOM        00    1238
                                         EXIT          54C     DVCHK         5DE     OVERFL        608     WERTIME       768
                                         WERPARMS      9A8
  IHOECOMH    1238     DE0
                                         IBCOM#       1278     IB081971     1278     FDIOCS#      1334     INTSWTCH     2000
  IHOCOMH2    2018     91D
                                         SEQDASD      232A
  IHOEFIOS    2938    1114
                                         FIOCS#       2938     FIOCSBEP     293E
  IHOFIOS2    3A50     5CC
  IHOERRM     4020     624
                                         ERRMON       4020     IHOERRE      4038
  IHOETRCH    4648     2A6
                                         IHOTRCH      4648     ERRTRA       4650
  IHOFCONI    48F0     2FD
                                         FQCONI#      48F0
  IHOFCONO    4BF0     4C2
                                         FQCONO#      4BF0
  IHOFCVTH    50B8     A07
                                         ADCON#       50B8     FCVAOUTP     5162     FCVLOUTP     51F2     FCVZOUTP     534E
                                         FCVIOUTP     56F6     FCVEOUTP     57E8     FCVCOUTP     57E8     INT6SWCH     5A40

  IHOFTEN     5AC0     198
                                         FTEN#        5AC0
  IHOUATBL    5C58     208
  IHOUOPT     5E60     338
  MAIN        6198    CED4
  AACOMP     13070     C5A
  AKZCAL     13CD0     686
  CADJUS     14358     3F8
  CCHECK     14750     17C
  CHEMIC     148D0     218
  CONSIN     14AE8     434
                                         CDTOTP      14EBC
  DIMENS     14F20     870
                                         DIMEN1      15648     HHCALC      15724
  DTTEST     15790     74A
  GEOIN      15EE0     B76
  OUTAPE     16A58     70E
  POSITV     17168     198
  PRINTS     17300    248C
                                         PRINTA      1967C     PRINTB      196C4     PRINTC      19718
  SHIFTN     19790     32A
```

NAME	ORIGIN	LENGTH	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION
SOURCE	19AC0	105E								
SOUSIN	1AB20	3E6								
STABIT	1AF08	55C								
STNCON	1B468	6A0								
TIMEX	1BB08	202								
UVFLUX	1BD10	8B4								
UVINTP	1C5C8	4C4								
UVZF	1CA90	A92								
WINDER	1D528	234								
WINDGR	1D760	D7C								
WINDIN	1E4E0	B8A								
WRITES	1F070	FBC								
			WRITEX	1FEA8	WRITEZ	1FF68				
WWFLUX	20030	6CA								
WWZF	20700	70A								
XYDIFF	20E10	706								
XYUTMS	21518	90C								
			XYUTM1	21D6C						
ZZDIFF	21E28	6D0								
ZZGRID	224F8	2BC								
IHOSLOG *	227B8	1D4								
			LOG10	227B8	ALOG10	227B8	IH$ALOGO	227B8	ALOG	227D0
			LOG	227D0	IH$ALOG	227D0				
IHOSATN2*	22990	1E8								
			ATAN2	22990	IH$ATAN2	22990	ATAN	229A4	IH$ATAN	229A4
IHOSSCN *	22B78	208								
			COS	22B78	IH$$COS	22B78	SIN	22B9A	IH$$SIN	22B9A
IHOSEXP *	22D80	1B0								
			EXP	22D80	IH$$EXP	22D80				
IHONAMEL*	22F30	C73								
			FRDNL#	22F30	FWRNL#	2360C				
IHOFRXPR*	23BA8	19B								
			FRXPR#	23BA8						
IHOSSQRT*	23D48	168								
			SQRT	23D48	IH$SQRT	23D48				
AADATA	23EB0	388								
CBLOCK	24238	20D00								

ENTRY ADDRESS 6198
TOTAL LENGTH 44F38

****MAIN DOES NOT EXIST BUT HAS BEEN ADDED TO DATA SET

NOTE:

The subprogram listing in the next two pages is an alternate code of subroutine WRITES. This version of subroutine WRITES replaces the previously listed routine when program IBMAQ-2 is to be executed on the EPA Research Triangle Park computer facility. This version of subroutine WRITES was modified from the original subroutine by Ms. Virginia Smiley of EPA.

```
CELT,SI RAPS*SHIR.WRITES,,,111645032413
      SUBROUTINE WRITES (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IUTM,JUTM
     *                    ,IFORM,JUNIT,RATIO,TITLE)
C
C.... THIS ROUTINE WRITES Q(ISIZE,JSIZE) ON UNIT=JUNIT IN FORM=IFORM.
C        (VERSION FOR UNIVAC MACHINE.)
C     IFORM = NO. OF COLUMES TO BE PRINTED ON ONE LINE.
C     II    = DUMMY ARRAY FOR TEMPORAL STORAGE
C
      DIMENSION RFMT1(3),RFMT2(3),RFMT3(3),TITLE(10),IFM(11)
     *          ,Q(IXMAX,IYMAX),IUTM(ISIZE),JUTM(JSIZE),II(130)
      DATA RFMT1/'(5X,5H','XUTM=,','30I4) '/
     *    ,RFMT2/'(5X,5H','   I=,','30I4/)'/
     *    ,RFMT3/'(1X,2I','4,1X, ','30I4) '/
     *    ,IFM/061,062,063,064,065,066,067,070,071,060,005/
      RETURN
C
C     ---------------------------------------------------
C
C *** PRINT HORIZONTAL SURFACE OF VARIABLE Q ***
C
      ENTRY WRITEX (Q,IXMAX,IYMAX,ISIZE,JSIZE,IFORM,JUNIT,RATIO,TITLE)
C
 8000 FORMAT (10A4,F7.2)
 8040 FORMAT ('    ..... PAGE = ',I5/)
C8040 FORMAT('1  ..... PAGE = ',I5,/5X,'IUTM=',30I4,/,5X,'   I=',40I3//)
C8050 FORMAT (1X,I4,I4,1X,30I4)
C
C.... DETERMINE DOMAIN OF ARRAY Q TO BE PRINTED.
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
C
C.... FIX VARIABLE FORMAT STATEMENTS
      IF (IFORM .EQ. 0) GO TO 500
      IF1=IFORM/10
      IF (IF1.GT.9) IF1=9
      IF2=IFORM-IF1*10
      IF (IF2.EQ.0) IF2=10
      IF (IF1.EQ.0) IF1=11
      IF3=120/IFORM
      IF (IF3.GT.9) IF3=9
      FLD(0,6,RFMT1(3))=IFM(IF1)
      FLD(0,6,RFMT2(3))=IFM(IF1)
      FLD(0,6,RFMT3(3))=IFM(IF1)
      FLD(6,6,RFMT1(3))=IFM(IF2)
      FLD(6,6,RFMT2(3))=IFM(IF2)
      FLD(6,6,RFMT3(3))=IFM(IF2)
      FLD(18,6,RFMT1(3))=IFM(IF3)
      FLD(18,6,RFMT2(3))=IFM(IF3)
      FLD(18,6,RFMT3(3))=IFM(IF3)
C
C.... DETERMINE NO. OF PAGES PER ARRAY Q TO BE PRINTED
      IPT=(ISIZE-1)/IFORM+1
C
C.... FIX RATIO OF PRINTED VALUE TO ACTUAL VALUE OF Q
      IF (RATIO.NE.0.0) GO TO 100
      RATIO=1000.0
      IF (IFORM .GE. 30) RATIO=100.
  100 CONTINUE
C
C.... PRINT THE SPECIFIED TITLE
      WRITE (JUNIT,8000) TITLE,RATIO
C
C *** ENTERING PRINTING LOOP ***
      DO 400 IP=1,IPT
      IA=(IP-1)*IFORM+IBEG
      IB=IA+IFORM-1
C
C.... FIX INDICES
      DO 200 I=IA,IB
      I1=I-IA+1
      II(I1)=I
      IF ((II(I1).GT.100).AND.(IFORM.GT.30)) II(I1)=II(I1)-100
  200 CONTINUE
C
C.... PRINT PAGE NO., INDICES & UTM COORDINATES
      IAX=IA-IBEG+1
      IBX=IAX+IFORM-1
```

```
C     WRITE (JUNIT,8040) IP,(IUTM(I),I=IAX,IBX),(II(I1),I1=IA,IB)
      WRITE (JUNIT,8040) IP
      WRITE (JUNIT,RFMT1)(IUTM(I),I=IAX,IBX)
      WRITE (JUNIT,RFMT2)(II(I1),I1=IA,IB)
C
C.... PRINT ARRAY Q
      DO 320 JJ=1,JSIZE
      J=(JEND+1)-JJ
      JUTMJ=JUTM(JSIZE+1-JJ)
      DO 300 I=IA,IB
      I1=I-IA+1
  300 II(I1)=Q(I,J)*RATIO
C     WRITE (JUNIT,8050) JUTMJ,J,(II(I1),I1=IA,IB)
      WRITE (JUNIT,RFMT3)JUTMJ,J,(II(I1),I1=IA,IB)
  320 CONTINUE
  400 CONTINUE
C
  500 CONTINUE
C
      RETURN
C
C     ------------------------------------------------
C....
C *** PRINT VERTICAL CROSS SECTION OF VARIABLE A FOR I=IMC AND J=JMC.
C
      ENTRY WRITEZ (A,Z,RATIO,IM,JM,KM,IMC,JMC,TITLA)
C....
      DIMENSION A(IM,JM,KM),Z(KM)
C
 8500 FORMAT (/' ** VERTICAL PROFILE OF ',A4,' AT I=',I3,'; J=',I3,'**')
 8502 FORMAT (/5X,'XUTM=',30I4)
 8504 FORMAT (5X,'   I=',30I4/4X,'Z',3X,'K')
 8506 FORMAT (/5X,'YUTM=',30I4)
 8508 FORMAT (5X,'   J=',30I4/4X,'Z',3X,'K')
 8510 FORMAT (1X,2I4,1X,30I4)
C
      WRITE (JUNIT,8500) TITLA,IMC,JMC
C
C.... PRINT Y-Z CROSS SECTION
      IA=1
      IB=IM
      IF (IM.LE.30) GO TO 605
      IA=(IM-30)/2+1
      IB=IA-1+30
  605 CONTINUE
      WRITE (JUNIT,8502) (IUTM(I),I=IA,IB)
      WRITE (JUNIT,8504) (I,I=IA,IB)
      J=JMC
      DO 620 KK=1,KM
      K=KM+1-KK
      KZ=Z(K)
      DO 610 I=IA,IB
  610 II(I)=A(I,J,K)*RATIO
      WRITE (JUNIT,8510) KZ,K,(II(I),I=IA,IB)
  620 CONTINUE
C
C.... PRINT X-Z CROSS SECTION
      JA=1
      JB=JM
      IF (JM.LE.30) GO TO 625
      JA=(JM-30)/2+1
      JB=JA-1+30
  625 CONTINUE
      WRITE (JUNIT,8506) (JUTM(J),J=JA,JB)
      WRITE (JUNIT,8508) (J,J=JA,JB)
      I=IMC
      DO 640 KK=1,KM
      K=KM+1-KK
      KZ=Z(K)
      DO 630 J=JA,JB
  630 II(J)=A(I,J,K)*RATIO
      WRITE (JUNIT,8510) KZ,K,(II(J),J=JA,JB)
  640 CONTINUE
C
      RETURN
C     DEBUG SUBCHK
      END
```

2. Auxiliary Program Listing

```
MEMBER NAME  EPAPTBL
C.... THIS PROGRAM READS IN EPA NEDS POINT SOURCE DATA AND PRINTS OUT
C     TABLE FOR EACH PLANT.
C     DIMENSION ID(200,140),ISEA(4)
      DIMENSION ISEA(4)
      LOGICAL*1 NAME(40),OWN
      DATA IUNIT/11/,JUNIT/6/
      NPLANT=0
1000  CONTINUE
100   READ (IUNIT,8000,END=1001) L1,L2,L3,ICD
8000  FORMAT (I2,I4,3X,I4,66X,I1)
      IF (ICD.EQ.9) GO TO 1001
      IF (ICD.EQ.6) GO TO 100
      IF (ICD.GE.3) GO TO 3000
      BACKSPACE IUNIT
      GO TO (101,102),ICD
101   CONTINUE
      NPOINT=0
      NPLANT=NPLANT+1
      NSTACK=0
      READ (IUNIT,8001) ISTATE,ICONTY,IPLANT,IUTM,(NAME(K),K=1,40),OWN
8001  FORMAT (I2,I4,3X,I4,4X,I2,2X,40A1,12X,A1)
      WRITE (JUNIT,8510) NPLANT,IUTM,ISTATE,ICONTY,IPLANT
     &        ,(NAME(K),K=1,40),OWN
8510  FORMAT (//I5,',    UTM=',I2,', STATE=',I2,', COUNTY=',I4
     &      ,', PLANT ID=',I4,8X,40A1,3X,';  OWNER=',A1/
     &      5X,'STACK',3X,'UTMX',3X,'UTMY',3X,'ZP',5X,'SO2',3X
     &      ,'SPACE',5X,'DIA',4X,'T',4X,'FLOW',3X,'ZR',4X,'S-1'
     &      ,2X,'S-2',2X,'S-3',2X,'S-4',3X,'HR',2X,'DAY',3X,'WK')
      IDUM=0
      IOCCM=0
102   CONTINUE
      NPOINT=NPOINT+1
      READ (IUNIT,8002) JSTATE,JCONTY,JPLANT,IPOINT,IXP,IYP,IZP,IDP,IT
     &        ,IFR,IPH,IPC1,IPC2,ICD
8002  FORMAT (I2,I4,3X,I4,I2,8X,I4,I5,I4,I3,I4,I7,I4,1X,2I2,20X,I1)
      IF (ICD.NE.2) GO TO 3000
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JCONTY.NE.ICONTY) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPLANT.NE.IPLANT) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
8520  FORMAT (' *** POSSIBLE ERROR AT : STATE,COUNTY,PLANT,CD',4I6)
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.3) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IERROR=0
103   CONTINUE
      READ (IUNIT,8003) JSTATE,JCONTY,JPLANT,JPOINT,(ISEA(K),K=1,4)
     &     ,NHR,NDAY,NWEEK,ISP,ISPACE,ICD
8003  FORMAT (I2,I4,3X,I4,I2,2X,5I2,I1,I2,7X,I7,26X,I3,6X,I1)
      IF (ICD.EQ.4) GO TO 104
      IF (IERROR.GT.0) GO TO 3000
      WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IERROR=1
      GO TO 103
104   CONTINUE
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JCONTY.NE.ICONTY) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPLANT.NE.IPLANT) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPOINT.NE.IPOINT) GO TO 3000
```

```
MEMBER NAME  EPAPTBL
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.5) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.6) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF ((IPC1.EQ.0).AND.(IPC2.EQ.0)) GO TO 201
      IQCOM=IQCOM+IQP
      IF (IPOINT.LT.IPC2) GO TO 1000
      IQP=IQCOM
      IQCOM=0
      ICOM=0
201   CONTINUE
      NSTACK=NSTACK+1
      XP=IXP
      XP=XP/10.
      YP=IYP
      YP=YP/10.
      DP=IDP
      DP=DP/10.
      SPACE=ISPACE
      SPACE=SPACE/10.
      WRITE (JUNIT,8530) NSTACK,XP,YP,IZP,IQP,SPACE,DP,IT,IER,IPH
     &      ,(ISEA(K),K=1,4),NHR,NDAY,NWEEK
8530  FORMAT (5X,I5,2F7.1,I5,I8,2F8.1,I5,I8,I5,2X,7I5)
      GO TO 1000
3000  WRITE (JUNIT,8550) ISTATE,ICONTY,IPLANT,ICD
8550  FORMAT (' ***** ERROR EXIT AT : STATE,COUNTY,PLANT,ICD =',4I6)
1001  CONTINUE
      RETURN
      END
```

```
MEMBER NAME  EPAPT
C.... THIS PROGRAM READS IN EPA NEDS POINT SOURCE DATA AND PRINTS OUT
C      TABLE FOR EACH PLANT.
C       IT ALSO CREATE DATA SET DSN=EPAPT1.DATA WHICH ELIMINATES
C       STACKS WITH ZERO EMISSION OF SO2.
C      DIMENSION IQ(200,140),ISEA(4)
      DIMENSION ISEA(4)
      LOGICAL*1 NAME(40),OWN,KKDATA(80)
      DATA IUNIT/11/,JUNIT/6/,KUNIT/12/
      NPLANT=0
      IOTCT=0
      NUMBER=0
      NOSO2=0
      IFLOW=0
1000  CONTINUE
100   READ (IUNIT,8000,END=1001) L1,L2,L3,ICD
8000  FORMAT (I2,I4,3X,I4,66X,I1)
      IF (ICD.EQ.9) GO TO 1001
      IF (ICD.EQ.6) GO TO 100
      IF (ICD.GE.3) GO TO 3000
      BACKSPACE IUNIT
      GO TO (101,102),ICD
101   CONTINUE
      NPOINT=0
      NPLANT=NPLANT+1
      NSTACK=0
      READ (IUNIT,8001) ISTATE,ICONTY,IPLANT,IUTM,(NAME(K),K=1,40),OWN
8001  FORMAT (I2,I4,3X,I4,4X,I2,2X,40A1,12X,A1)
      WRITE (JUNIT,8510) NPLANT,IUTM,ISTATE,ICONTY,IPLANT
     &       ,(NAME(K),K=1,40),OWN
8510  FORMAT (//I5,',    UTM=',I2,', STATE=',I2,', COUNTY=',I4
     &       ,', PLANT ID=',I4,8X,40A1,3X,';  OWNER=',A1/
     &       5X,'STACK',3X,'UTMX',3X,'UTMY',3X,'ZP',5X,'SO2',3X
     &       ,'SPACE',5X,'DIA',4X,'T',4X,'FLOW',3X,'ZR',4X,'S-1'
     &       ,2X,'S-2',2X,'S-3',2X,'S-4',3X,'HR',2X,'DAY',3X,'WK')
      ICOM=0
      IOCOM=0
102   CONTINUE
      NPOINT=NPOINT+1
      READ (IUNIT,8002) JSTATE,JCONTY,JPLANT,IPOINT,IXP,IYP,IZP,IDP,IT
     &       ,IFR,IPH,IPC1,IPC2,ICD
8002  FORMAT (I2,I4,3X,I4,I2,8X,I4,I5,I4,I3,I4,I7,I4,1X,2I2,20X,I1)
      IF (ICD.NE.2) GO TO 3000
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JCONTY.NE.ICONTY) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPLANT.NE.IPLANT) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
8520  FORMAT (' *** POSSIBLE ERROR AT : STATE,COUNTY,PLANT,CD',4I6)
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.3) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IERROR=0
103   CONTINUE
      READ (IUNIT,8003) JSTATE,JCONTY,JPLANT,JPOINT,(ISEA(K),K=1,4)
     &       ,NHR,NDAY,NWEEK,IOP,ISPACE,ICD
8003  FORMAT (I2,I4,3X,I4,I2,2X,5I2,I1,I2,7X,I7,26X,I3,6X,I1)
      IF (ICD.EQ.4) GO TO 104
      IF (IERROR.GT.0) GO TO 3000
      WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IERROR=1
      GO TO 103
```

```
MEMBER NAME  EPAPT
104   CONTINUE
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JCONTY.NE.ICONTY) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPLANT.NE.IPLANT) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JSTATE.NE.ISTATE) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF (JPOINT.NE.IPOINT) GO TO 3000
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.5) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      READ (IUNIT,8000) L1,L2,L3,ICD
      IF (ICD.NE.6) WRITE (JUNIT,8520) ISTATE,ICONTY,IPLANT,ICD
      IF ((IPC1.EQ.0).AND.(IPC2.EQ.0)) GO TO 201
      IQCOM=IQCOM+IQP
      IFLOW=IFLOW+IFR
      IF (IPOINT.LT.IPC2) GO TO 1000
      IQP=IQCOM
      IFR=IFLOW
      IQCOM=0
      IFLOW=0
      ICOM=0
201   CONTINUE
      NSTACK=NSTACK+1
      XP=IXP
      XP=XP/10.
      YP=IYP
      YP=YP/10.
      DP=IDP
      DP=DP/10.
      SPACE=ISPACE
      SPACE=SPACE/10.
      WRITE (JUNIT,8530) NSTACK,XP,YP,IZP,IQP,SPACE,DP,IT,IFR,IPH
     &      ,(ISEA(K),K=1,4),NHR,NDAY,NWEEK
8530  FORMAT (5X,I5,2F7.1,I5,I8,2F8.1,I5,I8,I5,2X,7I5)
      IF (IQP.EQ.0) GO TO 250
      NUMBER=NUMBER+1
      WRITE (KUNIT,8700) NPLANT,NSTACK,ISTATE,ICONTY,IPLANT,IUTM
     &      ,IXP,IYP,IZP,IQP,ISPACE,IDP,IT,IFR,IPH,(ISEA(K),K=1,4)
     &      ,NHR,NDAY,NWEEK
8700  FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
      IQTOT=IQTOT+IQP
      GO TO 1000
250   CONTINUE
      NOSO2=NOSO2+1
      GO TO 1000
3000  WRITE (JUNIT,8550) ISTATE,ICONTY,IPLANT,ICD
8550  FORMAT (' ***** ERROR EXIT AT : STATE,COUNTY,PLANT,ICD =',4I6)
1001  CONTINUE
      REWIND KUNIT
      QTOT=IQTOT
      WRITE (JUNIT,8710) NUMBER,QTOT,NOSO2
8710  FORMAT (1H1,' *** NUMBER OF POINT SOURCE =',I5,' ***'/
     &      '  *** TOTAL SO2 EMISSION =',E14.7,' (TONS/YR) ***'//
     &      '  *** NO. OF POINT SOURCE WITHOUT SO2 EMMISION =',I4,' ***'/
     &      /// '  *** REDUCED POINT SOURCE DATA ***'/
     &      '      ..... DSN=EPAPT1.DATA')
      DO 300 N=1,NUMBER
      READ (KUNIT,8750) (KKDATA(K),K=1,80)
      WRITE (JUNIT,8760) N,(KKDATA(K),K=1,80)
8750  FORMAT (80A1)
```

```
MEMBER NAME  FRAPT
8760  FORMAT (I5,'.  ',80A1)
300   CONTINUE
      RETURN
      END
```

```
MEMBER NAME  EPACLKTB
C.... THIS PROGRAM IS PROVIDED BY EPA. IT COPY DSN=EPACLKTB.DATA TO
C        DSN=EPACLK.DATA FOR DIRECT ACCESS READ.
C.... THE DATA SET IS CLARK'S TABLE. IT NEEDS TO COMPUTE UTM COORDINATES
C        OF A POINT IN ZONE II AS EXTENSION OF ZONE JJ.
      DIMENSION B1(8,8)
      REAL*8 B1
      DATA IUNIT/11/,JUNIT/6/
      DEFINE FILE 3(61,768,L,IJK)
      DO 10 JBLOCK=1,61
      READ(IUNIT,1000) ((B1(I,J),J=1,8),I=1,8)
      WRITE(3'JBLOCK) B1
   10 CONTINUE
      IJK=1
      DO 20 JBLOCK=1,61
      READ(3'JBLOCK) B1
      WRITE(JUNIT,2000) ((B1(I,J),J=1,8),I=1,8)
   20 CONTINUE
 1000 FORMAT(1X,F11.3,2F10.3,F8.3,2F6.3,2F5.3)
 2000 FORMAT(1X,F12.3,2F11.3,F9.3,2F7.3,2F6.3)
      STOP
      END
```

```
MEMBER NAME  EPAPT1
C.... THIS PROGRAM COMPUTES POINT SOURCE LOCATION ACCORDING
C       TO UTM ZONE 15.
C.... OUTPUT IS DSN=EPAPT2.DATA WITH ALL SOURCES LOCATION AS IN
C       UTM ZONE 15.
C     DIMENSION IQ(200,140),ISEA(4)
      DIMENSION ISEA(4)
      REAL*8 XP,YP,X,Y
      LOGICAL*1 NAME(40),CWN,KKDATA(80)
      DATA IUNIT/11/,JUNIT/6/,KUNIT/12/
      IQTOT=0
      NUMBER=0
      WRITE (JUNIT,8010)
8010  FORMAT (2X,'POINT  PLANT STACK UTM',5X,'UTMX'
     &        ,6X,'UTMY',6X,'X-15',6X,'Y-15',3X,'ZP',4X,'Q'//)
1000  CONTINUE
      READ (IUNIT,8000,END=1001) NPLANT,NSTACK,ISTATE,ICONTY,IPLANT
     &     ,IUTM,XP,YP,IZP,IQP,ISPACE,IDP,IT,IFR,IPH,(ISEA(K),K=1,4)
     &     ,NHR,NDAY,NWEEK
C8000 FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
8000  FORMAT (I4,2I2,2I4,I2,F4.1,F5.1,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
      IQTOT=IQTOT+IQP
      NUMBER=NUMBER+1
      IF (IUTM.EQ.15) GO TO 110
      IF (IUTM.EQ.0) GO TO 100
      IF (XP.EQ.0.) GO TO 100
      IF (YP.EQ.0.) GO TO 100
      X=XP*1000.
      Y=YP*1000.
      CALL GTGR (J,Y,X,0)
      X=X/1000.
      Y=Y/1000.
      GO TO 120
100   CONTINUE
      X=0.
      Y=0.
      GO TO 120
110   CONTINUE
      X=XP
      Y=YP
120   CONTINUE
      WRITE (JUNIT,8720) NUMBER,NPLANT,NSTACK,IUTM,XP,YP,X,Y,IZP,IQP
8720  FORMAT (2X,I5,',',2I6,I4,4F10.3,I5,I8)
8510  FORMAT (//I5,',    UTM=',I2,', STATE=',I2,', COUNTY=',I4
     &        ,'. PLANT ID=',I4,8X,40A1,3X,';  OWNER=',A1/
     &        5X,'STACK',3X,'UTMX',3X,'UTMY',3X,'ZP',5X,'SO2',3X
     &        ,'SPACE',5X,'DIA',4X,'T',4X,'FLOW',3X,'ZR',4X,'S-1'
     &        ,2X,'S-2',2X,'S-3',2X,'S-4',3X,'HR',2X,'DAY',3X,'WK')
      IXP=X*10.+0.5
      IYP=Y*10.+0.5
      WRITE (KUNIT,8700) NPLANT,NSTACK,ISTATE,ICONTY,IPLANT,IUTM
     &     ,IXP,IYP,IZP,IQP,ISPACE,IDP,IT,IFR,IPH,(ISEA(K),K=1,4)
     &     ,NHR,NDAY,NWEEK
8700  FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
      GO TO 1000
1001  CONTINUE
      QTOT=IQTOT
      WRITE (JUNIT,8710) NUMBER,QTOT
8710  FORMAT (/////' *** NUMBER OF POINT SOURCE =',I5,' ***'/
```

```
MEMBER NAME  EPAPT1
     &     '  *** TOTOL SO2 EMISSION =',E14.7,' (TONS/YR) ***')
300   CONTINUE
      WRITE (JUNIT,8750)
8750  FORMAT ('1  *** DATA SET DSN=EPAPT2.DATA ***' //)
      REWIND KUNIT
400   READ (KUNIT,8730,END=410) (KKDATA(K),K=1,80)
      WRITE (JUNIT,8740) (KKDATA(K),K=1,80)
      GO TO 400
410   CONTINUE
8730  FORMAT (80A1)
8740  FORMAT (5X,80A1)
      RETURN
      END
      SUBROUTINE GTGR(NARGS,CN,CE,IW)
C.... THIS SUBROUTINE IS PROVIDED BY EPA. IT COMPUTES UTM
C        COORDINATES FROM ONE ZONE TO ANOTHER.
      DIMENSION B1(8,8),NR(61)
      REAL*8 B1,A1,RN,RE,CNP,CEP,A,B,C,D,CN,CE,V1
      DEFINE FILE 3(61,768,L,IJK)
      DATA NR/20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,
     1         39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,
     2         58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,
     3         77,78,79,80/
      A1 = CN/100000.0
      IN = A1 + .5
      RN = IN * 100000
      A1 = CE/100000.0
      IE = A1 + .5
      RE = IE * 100000
      CNP = (CN-RN)*.00001
      CEP = (CE-RE)*.00001
      DO 40 I=1,61
      IF (NR(I).EQ.IN) GO TO 50
   40 CONTINUE
   50 IBLOCK = I
      READ(3'IBLOCK) B1
      IF (IW.EQ.1) GO TO 60
      IF (IW.EQ.0) GO TO 70
   60 IE = 10 - IE
      GO TO 80
   70 V1 = -1.0
      B1(IE,2) = B1(IE,2) * V1
      B1(IE,4) = B1(IE,4) * V1
      B1(IE,6) = B1(IE,6) * V1
      B1(IE,8) = B1(IE,8) * V1
   80 A =(B1(IE,5)+((CNP*B1(IE,7))-(CEP*B1(IE,8))))
      B =(B1(IE,6)+((CNP*B1(IE,8))+(CEP*B1(IE,7))))
      C =(B1(IE,3)+((CNP*A)-(CEP*B)))
      D =(B1(IE,4)+((CNP*B)+(CEP*A)))
      CN =(B1(IE,1)+((CNP*C)-(CEP*D)))
      CE =(500000.0+B1(IE,2)+((CNP*D)+(CEP*C)))
      RETURN
      END
```

```
MEMBER NAME   FPAPT2
C.... THIS PROGRAM COMPUTES PLUME RISES FOR ALL POINT SOURCES.
C        IT USES FIVE FORMULAE AS LISTED IN PRINT OUT.
      DIMENSION ISEA(4)
      DIMENSION ZR1(270),ZR2(270),ZR3(270),ZR4(270),ZR5(270)
       DIMENSION A(3),U1(3),U2(3),U3(3),U4(3),U5(3)
       DIMENSION PR1(3),PR2(3),PR3(3),PR4(3),PR5(3)
      DATA CP/.24/,RHO/1.29/
      DATA A/0.,0.5,1.0/
      DATA IUNIT/11/,JUNIT/6/
C
C     **CALCULATES PLUME RISE NOT ACCOUNTING FOR WIND SPEED**
C
      WRITE (JUNIT,8500)
8500  FORMAT ('1 *** PLUME RISE NORMALIZED BY WIND SPEED ***'//
     &      '  ZR1=(-0.029*VS*D+5.35*QH**0.5)/U....CARSON & MOSES'/
     &      '  ZR2=133.*(FF/U**3.)**0.27 ......CSANADY MODIFIED'/
     &      '  ZR3=0.047*QH**0.58/(U**0.70) ...CONCAWE--1'/
     &      '  ZR4=0.175*QH**0.50/(U**0.75) ...CONCAWE--2'/
     &      '  ZR5=0.414*QH**.444/(U**.694) ...CONCAWE--3'///
     &       '   N  ID  S    XP     YP    ZP        Q   D       TS'
     &       .'    FLOW     PH     ZR1     ZR2     ZR3     ZR4     ZR5'//)
      N=0
      CPRHO=CP*RHO*1000.
      PI4=4./3.14159
1000  CONTINUE
      READ (IUNIT,8000,END=1001) NPLANT,NSTACK,ISTATE,ICONTY,IPLANT
     &      ,IUTM,XP,YP,ZP,IQP,ISPACE,DP,TS,FR,PH,
     &      (ISEA(K),K=1,4),NHR,NDAY,NWEEK
8000  FORMAT (I4,2I2,2I4,I2,F4.1,F5.1,F4.0,I7,I3,F3.1,F4.0
     &      ,F7.0,F4.0,5I2,I1,I2)
      N=N+1
      ZP=ZP*0.3048
      DP=DP*0.3048
      IF (TS.EQ.0.) GO TO 100
      TS=(TS-32.)*5./9.
100   CONTINUE
      FR=FR*0.028317/60.
      PH=PH*0.3048
      IF (TS.LE.0.) GO TO 110
      IF (FR.LE.0.) GO TO 110
      VSD=FR*PI4/DP
      QH=FR*CPRHO*TS
      FF=9.81*TS*FR/273.
      ZR1(N)=(-0.029*VSD+5.35*(QH*0.001)**0.5)
      ZR2(N)=133.*FF**0.27
      ZR3(N)=0.047*QH**0.58
      ZR4(N)=0.175*QH**0.50
      ZR5(N)=0.414*QH**0.444
      GO TO 120
  110 CONTINUE
      ZR1(N)=0.
      ZR2(N)=0.
      ZR3(N)=0.
      ZR4(N)=0.
      ZR5(N)=0.
120   CONTINUE
      WRITE (JUNIT,8510) N,NPLANT,NSTACK,XP,YP,ZP,IQP,DP,TS,FR,PH,
     &      ZR1(N),ZR2(N),ZR3(N),ZR4(N),ZR5(N)
```

```
MEMBER NAME  EPAPT2
8510  FORMAT (1X,I3,I4,I3,F6.1,F7.1,F6.1,I8,F5.1,F7.1,F8.1,F5.1,
     &       5F7.1)
      GO TO 1000
1001  CONTINUE
       LM=N
C     **CALCULATES PLUME RISE FOR THREE VALUES OF WIND SPEED AND FOR
C        FOUR VALUES OF THE PARAMETER A**
C
      U=-2.
      DO 400 I=1,3
      U=U+3.
      WRITE (JUNIT,8600) U,(A(K),K=1,3)
8600  FORMAT ('1 *** PLUME RISE COMPUTED BY VARIOUS FORMULAE FOR U ='
     &      ,F3.1,' M/SEC ***'//6X,3(3X,13('-'),' A=',F5.2,1X,13('-'))/
     &   3X,'N',2X,3(6X,'ZR1',4X,'ZR2',4X,'ZR3',4X,'ZR4',4X,'ZR5',1X)/)
      DO 200 K=1,3
      UU=U+A(K)*EXP(-U)
      U1(K)=UU
      U2(K)=UU**(3.0*0.27)
      U3(K)=UU**0.70
      U4(K)=UU**0.75
      U5(K)=UU**0.694
200   CONTINUE
      DO 500 J=1,LM
      DO 300 K=1,3
      PR1(K)=ZR1(J)/U1(K)
      PR2(K)=ZR2(J)/U2(K)
      PR3(K)=ZR3(J)/U3(K)
      PR4(K)=ZR4(J)/U4(K)
      PR5(K)=ZR5(J)/U5(K)
300   CONTINUE
      WRITE (JUNIT,8610) J,(PR1(K),PR2(K),PR3(K),PR4(K),PR5(K),K=1,3)
  500 CONTINUE
  400 CONTINUE
8610  FORMAT (1X,I3,' *',3(3X,5F7.1))
      STOP
      END
```

```
MEMBER NAME  EPAPT2
C.... THIS PROGRAM MAPS POINT SOURCE EMISSION TO ONE SQ. KM. GRID.
      DIMENSION Q(200,140),ISFA(4)
      INTEGER*2 NPS(200,140),NPT
      DATA IUNIT/11/,JUNIT/6/,KUNIT/12/
      DATA Q/28000*0.0/,NPS/28000*0/
      QTOT=0
      NUMBER=0
      IXMAX=200
      IYMAX=140
      IXBEG=640
      IYBEG=4190
      XBEG=IXBEG
      YBEG=IYBEG
      XEND=XBEG+IXMAX
      YEND=YBEG+IYMAX
      ICXBEG=725
      ICYBEG=4252
      ICXEND=765
      ICYEND=4311
      CXBEG=ICXBEG
      CYBEG=ICYBEG
      CXEND=ICXEND
      CYEND=ICYEND
      WRITE (JUNIT,8010)
8010  FORMAT ('1    *** POINT SOURCE OUTSIDE AREA SOURCE MAP ***'
C    &       //'     ...... Q IN GRAM/SEC.  ..........'
     &        //2X,'POINT  PLANT STACK UTM',5X,'UTMX'
     &        ,6X,'UTMY',3X,'ZP',6X,'Q'//)
1000  CONTINUE
      READ (IUNIT,8000,END=1001) NPLANT,NSTACK,ISTATE,ICONTY,IPLANT
     &      ,IUTM,XP,YP,IZP,QP,ISPACE,IDP,IT,IFR,IPH,(ISFA(K),K=1,4)
     &      ,NHR,NDAY,NWEEK
C8000 FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
8000  FORMAT (I4,2I2,2I4,I2,F4.1,F5.1,I4,F7.0,2I3,I4,I7,I4,5I2,I1,I2)
C     QP=QP*0.02876664
C ....  .02876664 IS CONVERSION FACTOR FROM TONS/YEAR TO GRAM/SEC.
      NUMBER=NUMBER+1
      QTOT=QTOT+QP
      IF ((XP.GT.XEND).OR.(XP.LT.XBEG)) GO TO 100
      IF ((YP.GT.YEND).OR.(YP.LT.YBEG)) GO TO 100
      IXP=XP
      IYP=YP
      KX=IXP-IXBEG+1
      KY=IYP-IYBEG+1
      Q(KX,KY)=Q(KX,KY)+QP
      NPS(KX,KY)=NPS(KX,KY)+1
      GO TO 1000
100   CONTINUE
      WRITE (JUNIT,8720) NUMBER,NPLANT,NSTACK,IUTM,XP,YP,IZP,QP
8720  FORMAT (2X,I5,'.',2I6,I4,2F10.3,I5,F9.1)
      GO TO 1000
8510  FORMAT (//I5,',    UTM=',I2,', STATE=',I2,', COUNTY=',I4
     &       ,', PLANT ID=',I4,8X,40A1,3X,';  OWNER=',A1/
     &       5X,'STACK',3X,'UTMX',3X,'UTMY',3X,'ZP',5X,'SO2',3X
     &       ,'SPACE',5X,'DIA',4X,'T',4X,'FLOW',3X,'ZR',4X,'S-1'
     &       ,2X,'S-2',2X,'S-3',2X,'S-4',3X,'HR',2X,'DAY',3X,'WK')
8700  FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2)
1001  CONTINUE
```

```
MEMBER NAME  EPAPT3
      WRITE (JUNIT,8520)
8520  FORMAT (///'  *** POINT SOURCE EMISSION PER EACH ONE SQ. KM.'
     &       ,' GRID ***'//6X,'I',4X,'J UTMX  UTMY',7X,'Q',3X,'N'//)
      NPT=0
      DO 200 I=1,IXMAX
      DO 200 J=1,IYMAX
      IF (Q(I,J).LE.0.) GO TO 200
      IUTMX=I+IXBEG-1
      IUTMY=J+IYBEG-1
      NPT=NPT+NPS(I,J)
      WRITE (JUNIT,8530) I,J,IUTMX,IUTMY,Q(I,J),NPS(I,J)
8530  FORMAT (2X,3I5,I6,F10.1,I3)
200   CONTINUE
      WRITE (JUNIT,8540) NPT
8540  FORMAT (///' *** NUMBER OF POINT SOURCE IN AREA SOURCE MAP ='
     &         ,I5,' ***'//)
8710  FORMAT (/////' *** NUMBER OF POINT SOURCE =',I5,' ***'/
     &       ' *** TOTOL SO2 EMISSION =',E14.7,' (TONS/YR) ***')
300   CONTINUE
      CALL EPAMAP (Q,QTOT,NPS)
      RETURN
      END
```

```
MEMBER NAME  EPAPT4
      SUBROUTINE EPAMAP (Q,QTOT,NPS)
C.... THIS READS IN EPA NEDS AREA SOURCES DATA (TON/YEAR)
C.... IN UNIFORM 1-KM GRID AND CHOOSE A SUB-AREA FOR ANALYSIS.
C.... IXBEG,IYBEG=ORIGIN OF THE GRIDS IN UTM,IXMAX,IYMAX=AREA SIZE IN KM
C.... QA,Q(200,140)=AREA SOURCE STRENTH.
C.... IBEG,JBEG=ORIGIN OF THE SUB-AREA
C.... ISIZE,JSIZE= SIZE OF THE SUB-AREA.
C....
      DIMENSION Q(200,140), IQ(200),IXUTM(200),JYUTM(140),II(200)
     *          ,IA(200),JA(140)
      INTEGER*2 NPS(200,140),NPT
C     DATA Q/28000*0.0/,IUNIT/5/,JUNIT/6/,KUNIT/11/,IFORM/40/
      DATA IUNIT/5/,JUNIT/6/,KUNIT/11/,IFORM/20/
     *    ,IXBEG/640/,IYBEG/4190/,IXMAX/200/,IYMAX/140/
     *    ,IBEG/86/,JBEG/63/,ISIZE/40/,JSIZE/60/,DXA/1./,DYA/1./
      DIMENSION QQ(30,40),IB(10),JB(10),DXB(10),DYB(10)
      DATA IBM/3/,JBM/3/,IB/5,20,5,7*0/,JB/10,20,10,7*0/,IM/30/,JM/40/
     *    ,DXB/2.0,1.0,2.0,7*2.0/,DYB/2.0,1.0,2.0,7*2.0/
C....
      IA(1)=IBEG
      JA(1)=JBEG
C.... READ IN Q(200,140) AND QTOT.
C     REWIND KUNIT
C     DO 100 J=1,IYMAX
C     READ (KUNIT,8060) (Q(I,J),I=1,IXMAX)
 8060 FORMAT (10F7.5,10X)
  100 CONTINUE
C     READ (KUNIT,8070) QTOT
 8070 FORMAT (E12.6)
C....
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
      ITOT=ISIZE*JSIZE
C.... SUM THE TOTAL SOURCE QTOT FOR SUB-AREA.
      SQTOT=0.0
      NPT=0
      DO 200 J=JBEG,JEND
      DO 200 I=IBEG,IEND
      NPT=NPT+NPS(I,J)
  200 SQTOT=SQTOT+Q(I,J)
      RQTOT=SQTOT/QTOT
      WRITE (JUNIT,8030) ITOT,SQTOT,QTOT,RQTOT
 8030 FORMAT (' *** NUMBER OF AREA SOURCES = ',I10,' ***'/
     &        ' *** TOTAL SOURCE EMISSION & RATIO= ',3G13.5, ' ***')
      WRITE (JUNIT,8040) NPT
8040  FORMAT (/' *** NUMBER OF POINT SOURCES IN COMPUTATIONAL GRID ='
     &       ,I4,' ***'//)
C....
      DO 300 J=1,JSIZE
  300 JYUTM(J)=IYBEG+JBEG+J-2
      DO 310 I=1,ISIZE
  310 IXUTM(I)=IXBEG+IBEG+I-2
C     CALL WRITEQ(Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM,IFORM
C    *          ,JUNIT,II)
      WRITE (JUNIT,8510)
      DO 400 J=1,JSIZE
      DO 400 I=1,ISIZE
      Q(I,J)=Q(I,J)*0.02876664
```

```
MEMBER NAME  EPAPT4
      IF ((Q(I,J).GT.0.).AND.(Q(I,J).LT.1.)) Q(I,J)=1.0
C   .... 0.02876664 CONVERT TON/YR TO GRAM/SEC.
  400 CONTINUE
      CALL GRIDCN(Q,IXMAX,IYMAX,IA,JA,DXA,DYA,QQ,IM,JM,IB,JB,DXB,DYB,
     *            IBM,JBM,IXUTM,JYUTM,1)
      CALL WRITEQ(QQ,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II)
C     CALL WRITEQ (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM
C    &           ,IFORM,JUNIT,II)
      WRITE (JUNIT,8520)
8510  FORMAT (///' *** ABOVE PRINTING Q IN TONS/YEAR ***')
8520  FORMAT (///' *** ABOVE PRINTING Q IN GRAM/SEC. ***')
      RETURN
      END
      SUBROUTINE WRITEQ (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM
     *                  ,IFORM,JUNIT,II)
C.... THIS WRITES Q(ISIZE,JSIZE) ON UNIT=JUNIT IN FORM=IFORM.
C.... IFORM=NO. OF COLUMES TO BE PRINTED ON ONE LINE.
C....
      DIMENSION RFMT1(10),RFMT2(10),RFMT3(10)
      LOGICAL*1 FMT1(40),FMT2(40),FMT3(40),FM(8)
      EQUIVALENCE (FM(1),IFM1),(FM(5),IFM2)
     * ,(FMT1(1),RFMT1(1)),(FMT2(1),RFMT2(1)),(FMT3(1),RFMT3(1))
      DATA RFMT1/'(5X,','5HXU','TM=,','40I3',','    )',5*'    '/
     *    ,RFMT2/'(5X,','5H  ','  I=,','40I3',',/ )',5*'    '/
     *    ,RFMT3/'(1X,','2I4,','1X ,','40I3',','    )',5*'    '/
      DIMENSION Q(IXMAX,IYMAX),IXUTM(ISIZE),JYUTM(JSIZE) ,II(ISIZE)
     *          ,IQ(200)
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
      IF (IFORM .EQ. 0) GO TO 500
      IFM1=120/IFORM+240
      IFM2=IFORM/10+240
      FMT1(16)=FM(4)
      FMT2(16)=FM(4)
      FMT3(16)=FM(4)
      FMT1(13)=FM(8)
      FMT2(13)=FM(8)
      FMT3(13)=FM(8)
      IPT=(ISIZE-1)/IFORM+1
      RATIO=1000.0
      IF (IFORM .GE. 30) RATIO=100.
      IF (IFORM .GE. 40) RATIO=10.0
C     RATIO=10.0
      RATIO=1.0
      DO 310 IP=1,IPT
      IA=(IP-1)*IFORM+IBEG
      IB=IA+IFORM-1
      DO 320 I=IA,IB
      II(I)=I
      IF((II(I).GT. 100) .AND.(IFORM.GT. 30))II(I)=II(I)-100
  320 CONTINUE
      IAX=IA-IBEG+1
      IBX=IAX+IFORM-1
C     WRITE (JUNIT,8040) IP,(IXUTM(I),I=IAX,IBX),(II(I),I=IA,IB)
C8040 FORMAT('1  ***** PAGE = ',I5,/5X,'IUTM=',40I3,/,5X,'   I=',40I3//)
      WRITE (JUNIT,8040) IP
 8040 FORMAT ('1  ***** PAGE = ',I5/)
      WRITE (JUNIT,FMT1) (IXUTM(I),I=IAX,IBX)
```

```
MEMBER NAME  FRAPT4
      WRITE (JUNIT,FMT2) (II(I),I=IA,IB)
      DO 300 JJ=1,JSIZE
      J=(JEND+1)-JJ
      JUTM=JYUTM(JSIZE+1-JJ)
      DO 410 I=IA,IB
  410 IQ(I)=Q(I,J)*RATIO
C     WRITE (JUNIT,8050) JUTM,J,(IQ(I),I=IA,IB)
C8050 FORMAT (1X,I4,I4,1X,40I3)
      WRITE (JUNIT,FMT3) JUTM,J,(IQ(I),I=IA,IB)
  300 CONTINUE
  310 CONTINUE
      RETURN
  500 CONTINUE
C.... THIS WRITES Q ON DISK IN CARD IMAGE.
C....
C.... WRITE Q(I,J) ON DISK...
C     DO 400 J=1,IYMAX
C     WRITE (KUNIT,8060) (Q(I,J),I=1,IXMAX)
C8060 FORMAT (10F7.5,10X)
  400 CONTINUE
C     WRITE (KUNIT,8070) QTOT
C8070 FORMAT (F12.6)
      RETURN
      END
      SUBROUTINE GRIDCN(Q,IN,JN,IA,JA,DXA,DYA,QQ,IM,JM,IB,JB,DXB,DYB,
     *                  IBM,JBM,IXUTM,JYUTM,L)
C....
C.... L=1,THIS CONVERTS Q(IN,JN) IN UNIFORM GRID INTO QQ(IM,JM) IN NON
C.... UNIFORM GRID.
C.... L=2, CONVERT QQ INTO Q.
C.... IA,JA=LOCATION IN QQ GRID WHERE GRID SPACECHANGES, EXCEPT
C.... IA(1),JA(1) DENOTES THE ORIGIN OF QQ---QQ(1,1)=Q(IA(1),JA(1)).
C.... DX,DY=GRID SPACE WITH INITIAL POINT AT (IA,JA).
C....
      DIMENSION Q(IN,JN),QQ(IM,JM),IA(IM),JA(JM),DXB(IBM),DYB(JBM)
     *          ,IXUTM(IM),JYUTM(JM),IB(IBM),JB(JBM)
C....
      DXAI=1.0/DXA
      DYAI=1.0/DYA
      AREAI=DXAI*DYAI
      IBEG=IA(1)
      JBEG=JA(1)
      IXBEG=IXUTM(1)
      IYBEG=JYUTM(1)
C.... SET UP THE TABLE FOR INDICIES OF Q(I,J).
C....
      J1=0
      DO 20 J=1,JBM
      JINTVL=DYB(J)*DYAI+0.2
      JJEND=JB(J)
      DO 10 JJ=1,JJEND
      J1=J1+1
      IF (J1 .GE. JM) GO TO 20
      JA(J1+1)=JA(J1)+JINTVL
      JYUTM(J1+1)=JYUTM(J1)+JINTVL
   10 CONTINUE
   20 CONTINUE
      I1=0
```

```
MEMBER NAME  EPAPT4
      DO 40 I=1,IBM
      IINTVL=DXB(I)*DXAI+0.2
      IIEND=IB(I)
      DO 30 II=1,IIEND
      I1=I1+1
      IF (I1 .GE. IM) GO TO 40
      IA(I1+1)=IA(I1)+IINTVL
      IXUTM(I1+1)=IXUTM(I1)+IINTVL
   30 CONTINUE
   40 CONTINUE
C     DEBUG INIT(IA,JA)
C.... TRANSFORM Q(KI,KJ) TO QQ(I,J).
      DO 600 J=1,JM
      KJ=JA(J)
      JJJEND=JINTVL
      IF (J .LT. JM) JJJEND=JA(J+1)-JA(J)
      DO 500 I=1,IM
      KI=IA(I)
      IIIEND=IINTVL
      IF (I .LT. IM) IIIEND=IA(I+1)-IA(I)
C....
C.... AVERAGE THE Q(KI,KJ) TO GET QQ(II,JJ)
      QTOT=0.0
      DO 200 JJJ=1,JJJEND
      KJJ=KJ+JJJ-1
C....
      DO 100 III=1,IIIEND
      KII=KI+III-1
      QTOT=QTOT+Q(KII,KJJ)
  100 CONTINUE
  200 CONTINUE
      QQ(I,J)=QTOT*AREAI
  300 CONTINUE
  400 CONTINUE
  500 CONTINUE
  600 CONTINUE
      RETURN
C     DEBUG INIT (IA,JA,KI,KJ)
      END
```

```
MEMBER NAME  EPAPT5
C.... THIS PROGRAM COMBINES SMALL EMISSION SOURCES IN A PLANT
C       TO BE TREATED AS : 1) A SOURCE, IF (QS.GT.150): OR 2) PART
C       OF A LARGE SOURCE IN THE SAME PLANT, IF (QS.LE.150).
C.... QS IS COMBINEED SOURCE EMISSION OF STACK HAVE
C          Q LESS THAN OR EQUAL TO 150 TONS/YEAR.
C.... INPUT DATA SET=IUNIT, IT IS A REDUCED POINT SOURCE DATA FROM NEDS.
C.... THIS PROGRAM ALSO CREATES NEW DATA SETS FOR POINT SOURCE ON
C        I/O UNIT=KUNIT AND KUNIT1
      COMMON NPLANT(270),NSTACK(270),ISTATE(270),ICONTY(270)
     &       ,IPLANT(270),IUTM(270),XP(270),YP(270),ZP(270),IQP(270)
     &       ,SPACE(270),DP(270),TS(270),FR(270),PH(270),ISEA(270,4)
     &       ,NHR(270),NDAY(270),NWEEK(270),ZR4(270)
      DIMENSION MCOMB(20)
      LOGICAL*1 A(80)
      DATA CP/.24/,RHO/1.29/
      DATA IUNIT/11/,JUNIT/6/,KUNIT/12/,KUNIT1/13/
C *** FORMAT STATEMENTS ***
8000  FORMAT (I4,2I2,2I4,I2,F4.1,F5.1,F4.0,I7,2F3.1,F4.0,F7.0
     &      ,F4.0,5I2,I1,I2)
8600  FORMAT ('1 *****  POINT SOURCE DATA BEFORE SMALL EMISSION SOURCE'
     &      ,' BEING COMBINED *****'//'       .... ALL PARAMETERS ARE'
     &      ,' LISTED IN MGS UNIT, EXCEPT Q IN TONS/YEAR.'//
     &      3X,'N  ID  S ST   CT  ID UTM    X       Y      ZP      Q    SP'
     &      ,'   DP    TS  FLOW   PH --ISEA-- HR...     ZR4  '/)
8610  FORMAT (1X,I3,I4,2I3,I5,I4,I4,F6.1,F7.1,F6.1,I7,2F5.1,2F6.1
     &      ,F5.1,1X,4I2,1X,I2,I1,I2,F7.1)
8700  FORMAT ('1 ***** POINT SOURCE DATA SET WITH SMALL SOURCES BEING '
     &      ,'COMBINED *****'///
     &      3X,'N  ID  S ST   CT  ID     X       Y      ZP      Q    SP'
     &      ,'   DP    TS  FLOW   PH --ISEA-- HR...     ZR4  N(OLD)'/)
8900  FORMAT ('1 ***** DATA ON KUNIT, DSN=EPAPT3.DATA *****'//)
8910  FORMAT (80A1)
8940  FORMAT ('1 ***** DATA ON KUNIT1, DSN=EPAPT4.DATA *****'//)
8950  FORMAT (5X,80A1)
      CPRHO=CP*RHO*1000.
      U=4.**0.75
      WRITE (JUNIT,8600)
C *** READ POINT SOURCE DATA AND CONVERT TO MGS UNIT ***
C.... ZR4 IS PLUME RISE NORMALIZED BY WIND SPEED, CONCAWE FORMULA
C          IS USED IN THIS COMPUTATION.
      N=0
1000  CONTINUE
      N=N+1
      READ (IUNIT,8000,END=1001) NPLANT(N),NSTACK(N),ISTATE(N)
     &       ,ICONTY(N),IPLANT(N),IUTM(N),XP(N),YP(N),ZP(N),IQP(N)
     &       ,SPACE(N),DP(N),TS(N),FR(N),PH(N),(ISEA(N,K),K=1,4)
     &       ,NHR(N),NDAY(N),NWEEK(N)
      ZP(N)=ZP(N)*0.3048
      DP(N)=DP(N)*0.3048
      IF (TS(N).EQ.0.) GO TO 100
      TS(N)=(TS(N)-32.)*5./9.
100   CONTINUE
      FR(N)=FR(N)*0.028317/60.
      PH(N)=PH(N)*0.3048
      IF (TS(N).LE.0.) GO TO 110
      IF (FR(N).LE.0.) GO TO 110
      QH=FR(N)*CPRHO*TS(N)
      ZR4(N)=0.175*QH**0.50
```

```
MEMBER NAME  EPAPTS
      GO TO 120
110   CONTINUE
      ZR4(N)=0.
120   CONTINUE
      WRITE (JUNIT,8610) N,NPLANT(N),NSTACK(N),ISTATE(N)
     &      ,ICONTY(N),IPLANT(N),IUTM(N),XP(N),YP(N),ZP(N),IQP(N)
     &      ,SPACE(N),DP(N),TS(N),FR(N),PM(N),(ISFA(N,K),K=1,4)
     &      ,NHR(N),NDAY(N),NWEEK(N),ZR4(N)
      GO TO 1000
1001  CONTINUE
        NMAX=N-1
C *** PROCEED TO COMBINE SMALL SOURCES ***
      WRITE (JUNIT,8700)
      NSN=0
      NEW=0
C.... FIND THE SOURCES WITH SAME PLANT ID.
1002  CONTINUE
      JSTACK=0
      DO 140 K=1,20
      MCOMB(K)=0
140   CONTINUE
      MP=0
      M=0
      QS=0.
      HS=0.
      ZRS=0.
      NS=NSN+1
      IF (NS.GT.NMAX) GO TO 1100
      NID=NPLANT(NS)
      NSS=NS+1
      DO  200 N=NSS,NMAX
      IF (NPLANT(N).NE.NID) GO TO 210
200   CONTINUE
210   CONTINUE
      NSP=N-NS
      NSN=N-1
      IF (NSP.LE.1) GO TO 1003
C.... SEARCH FOR Q.LE.150 AND SUM IT TO QS.
      DO 220 N=NS,NSN
      IF (IQP(N).GT.150.) GO TO 220
      M=M+1
      MCOMB(M)=N
      QP=IQP(N)
      QS=QP+QS
      HS=QP*ZP(N)+HS
      ZRS=QP*ZR4(N)+ZRS
220   CONTINUE
      MP=M
      IF (MP.EQ.0) GO TO 300
      IF (QS.GT.150.) GO TO 400
C.... COMBINED SOURCE ADD TO OTHER STACK.
      HEFF=HS/QS+ZRS/(QS*U)
      MM=0
C.... NOW SEARCH FOR A STACK TO BE ADDED TO.
      ZDIFF=9999.
      DO 230 N=NS,NSN
      IF (IQP(N).LE.150) GO TO 230
      MM=MM+1
```

```
MEMBER NAME  EPARTS
      HOEFF=ZP(N)+ZR4(N)/U
      ZDIF=ABS(HOEFF-HEFF)
      IF (ZDIFF.LT.ZDIF) GO TO 230
      ZDIFF=ZDIF
      IPOINT=N
230   CONTINUE
      IF (MM.EQ.0) GO TO 500
C.... ADD QS TO Q(IPOINT)
      MP=MP+1
      MCOMB(MP)=IPOINT
      HS=HS+ZP(IPOINT)*IQP(IPOINT)
      ZRS=ZRS+ZR4(IPOINT)*IQP(IPOINT)
      QS=QS+IQP(IPOINT)
      ZP(IPOINT)=HS/QS
      ZR4(IPOINT)=ZRS/QS
      IQP(IPOINT)=QS
      GO TO 600
C *** CREATE A NEW POINT SOURCE DATA SET ***
C.... ONLY ONE STACK IN THIS PLANT, NO CHANGE.
1003  CONTINUE
      CALL WRITEP (NS,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,1,NS)
      GO TO 1002
C.... NO STACK IN A PLANT HAS Q LESS THAN 150, NO CHANGE
300   CONTINUE
      DO 310 N=NS,NSN
      CALL WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,1,N)
310   CONTINUE
      GO TO 1002
C.... COMBINED STACK TO BE CONSIDER AS A SOURCE
400   CONTINUE
      DO 410 N=NS,NSN
      IF (IQP(N).LE.150) GO TO 410
      CALL WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,1,N)
410   CONTINUE
420   CONTINUE
      N=MCOMB(1)
      IQP(N)=QS
      ZP(N)=HS/QS
      ZR4(N)=ZRS/QS
      CALL WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,MP,MCOMB)
      GO TO 1002
C.... NO SOURCE IN THIS PLANT HAS Q LARGER THAN 150.
500   CONTINUE
      GO TO 420
C.... QS ADDED TO Q(IPOINT)
600   CONTINUE
      DO 620 N=NS,NSN
      IF (IQP(N).LE.150) GO TO 620
      IF (N.EQ.IPOINT) GO TO 610
      CALL WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,1,N)
      GO TO 620
610   CONTINUE
      CALL WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,MP,MCOMB)
620   CONTINUE
      GO TO 1002
1100  CONTINUE
      REWIND KUNIT
      REWIND KUNIT1
```

```
MEMBER NAME  FRAPT5
      WRITE (JUNIT,8900)                                                00001750
700   CONTINUE                                                          00001760
      READ (KUNIT,8910,END=800) (A(K),K=1,80)                           00001770
      WRITE (JUNIT,8950) (A(K),K=1,80)                                  00001780
      GO TO 700                                                         00001790
800   CONTINUE                                                          00001800
      WRITE (JUNIT,8940)                                                00001810
810   CONTINUE                                                          00001820
      READ (KUNIT1,8910,END=1110) (A(K),K=1,80)                         00001830
      WRITE (JUNIT,8950) (A(K),K=1,80)                                  00001840
      GO TO 810                                                         00001850
1110  CONTINUE                                                          00001860
      STOP                                                              00001870
      END                                                               00001880
      SUBROUTINE WRITEP (N,JSTACK,NEW,JUNIT,KUNIT,KUNIT1,MQ,MOLD)       00001890
      COMMON NPLANT(270),NSTACK(270),ISTATE(270),ICONTY(270)            00001900
     &       ,IPLANT(270),IUTM(270),XP(270),YP(270),ZP(270),IQP(270)    00001910
     &       ,SPACE(270),DP(270),TS(270),FR(270),PH(270),ISEA(270,4)    00001920
     &       ,NHR(270),NDAY(270),NWEEK(270),ZR4(270)                    00001930
      DIMENSION MOLD(MQ)                                                00001940
      NEW=NEW+1                                                         00001950
      JSTACK=JSTACK+1                                                   00001960
      WRITE (JUNIT,8710) NEW,NPLANT(N),JSTACK,ISTATE(N)                 00001970
     &      ,ICONTY(N),IPLANT(N),XP(N),YP(N),ZP(N)                      00001980
     &      ,IQP(N),SPACE(N),DP(N),TS(N),FR(N),PH(N)                    00001990
     &      ,(ISEA(N,K),K=1,4),NHR(N),NDAY(N),NWEEK(N)                  00002000
     &      ,ZR4(N),(MOLD(K),K=1,MQ)                                    00002010
      ID=1                                                              00002011
      IXP=XP(N)*10.                                                     00002020
      IYP=YP(N)*10.                                                     00002030
      IZP=ZP(N)*10.                                                     00002040
      ISPACE=SPACE(N)*10.                                               00002050
      IDP=DP(N)*10.                                                     00002060
      ITS=TS(N)*10.                                                     00002070
      IFR=FR(N)*10.                                                     00002080
      IPH=PH(N)*10.                                                     00002090
      IZR4=ZR4(N)*10.                                                   00002100
      ZPH=ZP(N)+PH(N)                                                   00002110
      IF (XP(N).LE.0.0) GO TO 100                                       00002111
      IF (YP(N).LE.0.0) GO TO 100                                       00002112
      IF ((ZP(N).LE.0.).AND.(PH(N).LE.0.)) GO TO 100                    00002113
      IF ((ZR4(N).LE.0.).AND.(PH(N).LE.0.)) GO TO 100                   00002114
      WRITE (KUNIT1,8820) NEW,XP(N),YP(N),ZPH,IQP(N),ZR4(N)             00002120
      GO TO 110                                                         00002121
100   CONTINUE                                                          00002122
      WRITE (KUNIT1,8830) NEW,XP(N),YP(N),ZPH,IQP(N),ZR4(N)             00002123
     &                   ,ID,XP(N),YP(N),ZPH,IQP(N),ZR4(N)              00002124
110   CONTINUE                                                          00002125
      WRITE (KUNIT1,8810) NPLANT(N),JSTACK,ISTATE(N)                    00002130
     &      ,ICONTY(N),IPLANT(N),IUTM(N),IXP,IYP,IZP                    00002140
     &      ,IQP(N),ISPACE,IDP,ITS,IFR,IPH                              00002150
     &      ,(ISEA(N,K),K=1,4),NHR(N),NDAY(N),NWEEK(N)                  00002160
     &      ,NEW                                                        00002170
8710  FORMAT (1X,I3,I4,2I3,I5,I4,F6.1,F7.1,F6.1,I7,2F5.1,2F6.1          00002180
     &      ,F5.1,1X,4I2,1X,I2,I1,I2,F7.1,' *',10I3/100X,10I3)          00002190
8810  FORMAT (I4,2I2,2I4,I2,I4,I5,I4,I7,2I3,I4,I7,I4,5I2,I1,I2,5X,I3)   00002200
8820  FORMAT (I5,3F7.1,I7,F7.1)                                         00002210
8830  FORMAT (2(I5,3F7.1,I7,F7.1))                                      00002211
```

```
MEMBER NAME  FPARTS
         RETURN                                                         00002220
         END                                                            00002230
```

```
MEMBER NAME  EPAPT6
C.... THIS PROGRAM PRINT OUT POINT SOURCES DATA ACCORDING TO
C         VARIOUS SEQUENCE.
C.... INPUT DATA SET IS REDUCED POINT SOURCE DATA, DSN=EPAPT5.DATA
C      IT ALSO CREATE DATA SET DSN=EPAPT6.DATA FOR MODEL INPUT
      DIMENSION NP(200),XP(200),YP(200),ZP(200),QP(200),ZR(200)
      DIMENSION NP1(200),XP1(200),YP1(200),ZP1(200),QP1(200),ZR1(200)
      DIMENSION IGP(200)
      LOGICAL*1 YES,A(80),IDD(200),ID1(200)
      DATA IUNIT/11/,JUNIT/6/,KUNIT/12/
      DATA IXMAX/200/,IYMAX/140/,XBEG/640./,YBEG/4190./
     &    ,CXBEG/725./,CYBEG/4252./,CXEND/765./,CYEND/4311./
      DATA XCC/745./,YCC/4282./
       DATA YES/'*'/,IDD/200*' '/
      REAL*4 FMTDS(10)/'(I5,','5F10','.1) ',7*'    '/
      WRITE (KUNIT,8010) FMTDS
8010  FORMAT (10A4)
      XEND=XBEG+IXMAX
      YEND=YBEG+IYMAX
      M=1
      QTOT=0.
1000  CONTINUE
      READ (IUNIT,8050,END=1001) NP(M),XP(M),YP(M),ZP(M),QP(M),ZR(M),ID
8050  FORMAT (I5,3F7.1,F7.0,F7.1,I5)
      IF (M.NE.NP(M)) GO TO 2000
      QTOT=QTOT+QP(M)
      IF (ID.EQ.1) IDD(M)=YES
      M=M+1
      GO TO 1000
1001  CONTINUE
      NMAX=M-1
C.... ARRANGE SOURCE ACCORDING TO EMISSION
      DO 110 K=1,NMAX
      QMAX=0.
      DO 100 N=1,NMAX
      IF (QP(N).LE.QMAX) GO TO 100
      QMAX=QP(N)
      MP=N
100   CONTINUE
      NP1(K)=NP(MP)
      XP1(K)=XP(MP)
      YP1(K)=YP(MP)
      ZP1(K)=ZP(MP)
      ZR1(K)=ZR(MP)
      ID1(K)=IDD(MP)
      QP1(K)=QP(MP)
      QP(MP)=0.
110   CONTINUE
      WRITE (JUNIT,8100)
8100  FORMAT ('1*** POINT SOURCE ARRANGED BY EMISSION '//
     &     '    .... (*) INDICATED MISSING DATA IN ORIGINAL FILE'//
     &     '    N',8X,'XP',8X,'YP',8X,'ZP',8X,'QP',8X,'ZR',3X,'NN')
      DO 120 N=1,NMAX
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N),N
     &     ,ID1(N)
8110  FORMAT (I5,5F10.1,I5,3X,A1)
120   CONTINUE
C.... GROUP SOURCE ACCORDING TO REGION
      INCGD=0
```

```
MEMBER NAME  EPAPT6
      INAGD=0
      QINC=0.
      QINA=0
      WRITE (JUNIT,8200)
8200  FORMAT ('1*** POINT SOURCE ARRANGED BY REGION ***'//
     &     '     N',8X,'XP',8X,'YP',8X,'ZP',8X,'QP',8X,'ZR',3X,'NN')
      DO 200 N=1,NMAX
      IGP(N)=3
      IF ((XP1(N).LT.XBEG).OR.(XP1(N).GT.XEND)) GO TO 200
      IF ((YP1(N).LT.YBEG).OR.(YP1(N).GT.YEND)) GO TO 200
      INAGD=INAGD+1
      QINA=QINA+QP1(N)
      IGP(N)=2
      IF ((XP1(N).LT.CXBEG).OR.(XP1(N).GT.CXEND)) GO TO 200
      IF ((YP1(N).LT.CYBEG).OR.(YP1(N).GT.CYEND)) GO TO 200
      INCGD=INCGD+1
      IGP(N)=1
      QINC=QINC+QP1(N)
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,INCGD,ID1(N)
C     WRITE (KUNIT,FMTDS) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
200   CONTINUE
      I=INCGD
      DO 210 N=1,NMAX
      IF (IGP(N).NE.2) GO TO 210
      I=I+1
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,I,ID1(N)
210   CONTINUE
      DO 220 N=1,NMAX
      IF (IGP(N).NE.3) GO TO 220
      I=I+1
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,I,ID1(N)
220   CONTINUE
      QTOTR=1
      QINCR=QINC/QTOT
      QINAR=QINA/QTOT
      WRITE (JUNIT,8210) NMAX,QTOT,QTOTR,INCGD,QINC,QINCR
     &      ,INAGD,QINA,QINAR
8210  FORMAT (///9X,'N',11X,'Q',5X,'RATIO'/
     &      5X,I5,F12.1,F10.4,3X,'--- TOTAL'/
     &      5X,I5,F12.1,F10.4,3X,'--- SOURCE WITHIN COMP. REGION'/
     &      5X,I5,F12.1,F10.4,3X,'--- SOURCE WITHIN AREA SOURCE MAP')
C.... GROUP SOURCE ACCORDING TO X , Y
      WRITE (JUNIT,8300)
8300  FORMAT ('1*** POINT SOURCE ARRANGED BY (I,J) OF TOTAL AREA ***'//
     &     '     N',8X,'XP',8X,'YP',8X,'ZP',8X,'QP',8X,'ZR',3X,'NN')
      M=0
      DO 310 J=1,IYMAX
      IYY=(J-1)+YBEG
      DO 310 I=1,IXMAX
      IXX=(I-1)+XBEG
      DO 300 N=1,NMAX
      IF (IGP(N).EQ.3) GO TO 300
      IXP=XP1(N)
      IF (IXX.NE.IXP) GO TO 300
      IYP=YP1(N)
```

```
MEMBER NAME  FPAPT6
      IF (IYY.NE.IYP) GO TO 300
      M=M+1
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,M,ID1(N)
      IF (IGP(N).EQ.2) GO TO 300
      WRITE (KUNIT,FMTDS) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
300   CONTINUE
310   CONTINUE
      DO 330 N=1,NMAX
      IF (IGP(N).NE.3) GO TO 330
      M=M+1
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,M,ID1(N)
330   CONTINUE
C.... ARRANGE SOURCE ACCORDING TO DISTANCE TO XCC,YCC
      WRITE (JUNIT,8500) XCC,YCC
8500  FORMAT ('1*** POINT SOURCE ARRANGED BY DISTANCE TO (XCC,YCC) ***'/
     &      20X,'(XCC,YCC)=(',F5.1,',',F6.1,')'//
     &  '     N',8X,'XP',8X,'YP',8X,'ZP',8X,'QP',8X,'ZR',3X,'NN',8X,'S')
      M=0
      DO 510 K=1,150
      XK=K
       XKM1=XK-1.
      DO 500 N=1,NMAX
       U=XCC-XP1(N)
       V=YCC-YP1(N)
       DS=SQRT(U*U+V*V)
       IF ((DS.GE.XK).OR.(DS.LT.XKM1)) GO TO 500
      M=M+1
      WRITE (JUNIT,8510) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N)
     &      ,ZR1(N),M,ID1(N),K
8510  FORMAT (I5,5F10.1,I5,3X,A1,I5)
500   CONTINUE
510   CONTINUE
      DO 520 N=1,NMAX
      IF (IGP(N).NE.3) GO TO 520
      M=M+1
      WRITE (JUNIT,8110) NP1(N),XP1(N),YP1(N),ZP1(N),QP1(N),ZR1(N)
     &      ,M,ID1(N)
520   CONTINUE
C.... PRINT DATA ON UNIT=KUNIT
      WRITE (JUNIT,8600)
      REWIND KUNIT
600   READ (KUNIT,8610,END=610) (A(I),I=1,80)
      WRITE (JUNIT,8620) (A(I),I=1,80)
      GO TO 600
610   CONTINUE
8600  FORMAT ('1  *** DATA ON UNIT=KUNIT ***'///)
8610  FORMAT (80A1)
8620  FORMAT (5X,80A1)
2000  CONTINUE
      RETURN
      END
```

```
MEMBER NAME  EPASTA
C.... THIS PROGRAM IS PROVIDED BY EPA. IT COMPUTES UTM COORDINATES
C       FROM GIVEN LAT. AND LONG. COORDINATES.
C.... INPUT IS CORDINATES OF RAPS STATION IN LAT. AND LONG.
C.... UNIT 3 IS CLARK'S TABLE.
C.... FILEDEF 03 DISK (XTENT 61 PERM RECFM FA BLOCK 768)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XRAMS(30),YRAMS(30)
      REAL*4 FMTDS(10)/'(3(I','5,F9','.3,F','10.3','))  ',5*'    '/
      LOGICAL*1 KKDATA(80)
       DATA IUNIT/11/,JUNIT/6/,KUNIT/12/
       NS=0
      WRITE (JUNIT,2)
    2 FORMAT('    LATITUDE          LONGITUDE       ZONE     VERTICAL UTM    HOR
     *IZONTAL UTM     CODE'/)
    1 READ(IUNIT,10,END=21)XAT1,XAT2,XAT3,XNG1,XNG2,XNG3
   10 FORMAT(6F3.0)
      NS=NS+1
      XLAT=(XAT3+60.*XAT2+3600.*XAT1)*1.0D5
      XLNG=(XNG3+60.*XNG2+3600.*XNG1)*(-1.0D5)
      CALL CED009(XLNG,XLAT,4,MZON,FMNOR,FMEAS,MERR)
      IF(MZON.EQ.15)GO TO 15
      CALL GTCR(1,FMNOR,FMEAS,0)
   15 WRITE(JUNIT,11)XAT1,XAT2,XAT3,XNG1,XNG2,XNG3,MZON,FMNOR
     &      ,FMEAS,MERR
      XRAMS(NS)=FMEAS*0.001
      YRAMS(NS)=FMNOR*0.001
9000  FORMAT (I4,6F6.1,I4,-3P2F10.3)
   11 FORMAT(3F4.0,4X,3F4.0,4X,I3,5X,F10.0,6X,F10.0,7X,I2)
      GO TO 1
21    CONTINUE
      NSTOT=NS
      WRITE (KUNIT,8010) NSTOT
8010  FORMAT (I5)
      WRITE (KUNIT,8100) FMTDS
8100  FORMAT (10A4)
      WRITE (KUNIT,FMTDS) (NS,XRAMS(NS),YRAMS(NS),NS=1,NSTOT)
       WRITE (JUNIT,8050)
8050   FORMAT ('1  *** DATA SET DSN=EPARAMS.DATA ***'//)
       REWIND KUNIT
200   READ (KUNIT,8200,END=300) (KKDATA(K),K=1,80)
8200  FORMAT (80A1)
      WRITE (JUNIT,8210) (KKDATA(K),K=1,80)
8210   FORMAT (5X,80A1)
      GO TO 200
300   CONTINUE
      STOP
      END
      SUBROUTINE CED009(XLNG,XLAT,IND1,      MZON,FMNOR,FMEAS,MERR)
C     MODIFIED BY HAWS. STI, 12/72 FOR ONLY 1 SPHEROID,I.E. WHERE IND2=2
C     GEODETIC POSITION TO UNIVERSAL TRANSVERSE MERCATOR COORDINATES
C     RECH/COMPUTER CONCEPTS
      COMMON/CORD/CCORD
      REAL*8 CCORD(16,7)   ,FMPHI,FDELTA
      REAL*8 DELTA,ALPHA,DELTA0,DELTA1,BETA
      REAL*8 GAMMA,DDELTA,FTERM
      REAL*8  FIMLAT,FIMLNG,FILNG,FILAT,FMN,FMEAS
      REAL*8 FMNOR,FMEAS,XLNG,XLAT
      MERR=0
```

```
MEMBER NAME  FPASTA
      IF(IND1)1,1,2
    1 MERR=1
      RETURN
    2 IF(IND1-5)3,1,1
    3 GO TO (5,6,7,8),IND1
    5 FILNG=XLNG*78683.7792D0
      FILAT=XLAT*78683.7792D0
      GO TO 13
    6 FILNG=XLNG *.38146972656D0
      FILAT=XLAT *.38146972656D0
      GO TO 13
    7 FILNG=XLNG*2.062648062D0
      FILAT=XLAT*2.062648062D0
      GO TO 13
    8 FILNG = XLNG
      FILAT = XLAT
   13 IF(FILAT)14,15,15
   14 FIMLAT=-FILAT
      GO TO 16
   15 FIMLAT=FILAT
   16 CONTINUE
      IF(FIMLAT-2.88D10) 18,18,17
   17 MERR=2
      RETURN
   18 IF(FILNG)19,20,20
   19 FIMLNG=-FILNG
      GO TO 21
   20 FIMLNG=FILNG
   21 CONTINUE
      IF(FIMLNG-6.48D10) 24,23,22
   22 MERR=3
      RETURN
   23 MZON=1
      GO TO 25
   24 MZON = (FILNG+6.696D10)/2.16D9
   25 CONTINUE
  600 CONTINUE
      IF(FIMLAT-72.D8) 201,202,601
  601 CONTINUE
      IF(FIMLAT-144.D8) 202,203,602
  602 CONTINUE
      IF(FIMLAT-198.D8) 203,204,603
  603 CONTINUE
      IF(FIMLAT-234.D8) 204,205,604
  604 CONTINUE
      IF(FIMLAT-262.8D8) 205,206,605
  605 CONTINUE
      IF(FIMLAT-277.2D8) 206,207,207
  201 J=1
      FMPHI = 360.D7
      GO TO 210
  202 J=2
      FMPHI = 108.D8
      GO TO 210
  203 J=3
      FMPHI = 180.D8
      GO TO 210
  204 J=4
```

```
MEMBER NAME  EPASTA
      EMPHI   216.D8
      GO TO 210
  205 J=5
      EMPHI = 252.D8
      GO TO 210
  206 J=6
      EMPHI = 270.D8
      GO TO 210
  207 J=7
      EMPHI   282.6D8
  210 DELTA = (FIMLAT-EMPHI)*.0000100
      I=6
      ALPHA=COORD(7,J)   +COORD(8,J)   *DELTA
  211 ALPHA=COORD(I,J)   +ALPHA*DELTA
      I=I-1
      IF(I)213,213,211
  213 DELTAQ=DELTA*ALPHA
      FTERM = MZON-31
      DELTAL=(216.D7*  FTERM    +108.D7)-FILNG
      DELTAL=DELTAL*4848136811.D-20
      ALPHA=COORD(15,J)   +COORD(16,J)   *DELTAQ
      BETA=COORD(16,J)   *DELTAL
      I=14
  215 GAMMA=COORD(I,J)   +ALPHA*DELTAQ-BETA*DELTAL
      DDELTA=ALPHA*DELTAL+BETA*DELTAQ
      ALPHA=COORD(I-1,J)   +GAMMA*DELTAQ-DDELTA*DELTAL
      BETA=GAMMA*DELTAL+DDELTA*DELTAQ
      I=I-2
      IF(I-10)220,220,215
  220 GAMMA=COORD(10,J)   +ALPHA*DELTAQ-DELTAL*BETA
      DDELTA=ALPHA*DELTAL+BETA*DELTAQ
      EMN   COORD(9,J)   +GAMMA*DELTAQ-DDELTA*DELTAL
      EMEAS    5.005 -(GAMMA*DELTAL+DDELTA*DELTAQ)
 5000 FORMAT(2D25.16)
      IF(XLAT)225,224,224
  225 EMNOR = 1.0D7-EMN
      GO TO 227
  224 EMNOR    EMN
  227 RETURN
      END
      BLOCK DATA
      COMMON / CORD/COORD
      REAL*8 COORD(16,7)
      REAL*8 CORD12(16),CORD22(16),CORD32(16),CORD42(16),CORD52(16),
     1     CORD62(16),CORD72(16)
      EQUIVALENCE
     1     (COORD(1,1)   , CORD12(1)) ,(COORD(1,2)   ,CORD22(1)),
     2     (COORD(1,3)   , CORD32(1)) ,(COORD(1,4)   ,CORD42(1)),
     3     (COORD(1,5)   , CORD52(1)) ,(COORD(1,6)   ,CORD62(1)),
     4     (COORD(1,7)   , CORD72(1))
      DATA CORD12/ .4890603699D-05, .0211783470D-10,
     1.0206094087D-15, .0021352709D-20, .0013344641D-25,
     2.0002154192D-30, .0001051211D-35, .0000219788D-40,
     31105343.1000D0,.6279435439D07,-.0545206260D07,-.0990165054D07,
     4   .0221578063D07, .0222595275D07, -.0086799370D07,
     5 -.0054352765D07 /
      DATA  CORD22/ .5569679060D-05, .0787424577D-10,
     1.0366618535D-15, .0107351529D-20, .0043800764D-25,
```

```
MEMBER NAME  FPASTA
     2.0016151478D-30, .0006562645D-35, .0002627691D-40,
     33318605.326D0,.5526156860D07, -.1381539215D07, -.0464043656D07,
     4 .0406929968D07, -.0023737423D07, -.0091961450D07,
     5.0037826746D07 /
      DATA CORD32 / .7521184517D-05, .2184990662D-10,
     1.1135593294D-15, .0576965638D-20, .0323544317D-25,
     2.0186820832D-30, .0111283189D-35, .0067606710D-40,
     35538411.779D0,.4106355473D07, -.1572825396D07, .0118047314D07,
     4 .0195232064D07, -.0109480462D07, .0014859044D07,
     5 .0015131135D07 /
      DATA CORD42/ .9679783520D-05, .4077989432D-10,
     1 .2659560366D-15, .1833896462D-20, .1361507835D-25,
     2 .1050208779D-30, .0833567868D-35, .0675334139D-40,
     3 6651204.6D0,.3195949863D07,  -.1383886979D07, .0266102296D07,
     4 .0058104357D07, -.0067036894D07, .0025282671D07,
     5 -.0002169360D07 /
      DATA CORD52 / 1.4163711960D-05, .9448163773D-10,
     1 .8939144732D-15, .9299525223D-20, 1.0336559489D-25,
     2 1.1963321013D-30, 1.4242120367D-35, 1.7308124042D-40,
     3 7765697.7D0,.2187148430D07, -.1027623620D07, .0279208157D07,
     4 -.0025458841D07, -.0018513695D07, .0012562355D07,
     5 -.0004359447D07 /
      DATA CORD62 / 1.8723215513D-05, 1.6953882434D-10,
     1 2.1175151210D-15, 2.9398158655D-20, 4.3557493117D-25,
     2 6.7218060066D-30, 10.6695415012D-35, 17.2886054901D-40,
     3 8323452.6D0,.1655376293D07, -.0799485147D07, .0238924408D07,
     4 -.0039827738D07, -.0003223536D07, .0005528952D07,
     5 -.0002522388D07 /
      DATA CORD72/ 2.4310955817D-05, 2.8981418519D-10,
     1 4.6980991563D-15, 8.5098855196D-20, 16.4479412680D-25,
     2 33.1014282702D-30, 68.5329893109D-35, 144.8462561026D-40,
     3 8714033.1D0, .1275252086D07, -.0624825325D07, .0195643581D07,
     4 -.0039646076D07, .0002575786D07, .0002080265D07,
     5 -.0001257142D07 /
      END
      SUBROUTINE GTGR(NARGS,CN,CE,IW)
      DIMENSION B1(8,8),NR(61)
      REAL*8 B1,A1,RN,RE,CNP,CEP,A,B,C,D,CN,CE,V1
      DEFINE FILE 3(61,768,L,IJK)
      DATA NR/20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,
     1          39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,
     2          58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,
     3          77,78,79,80/
      A1 = CN/100000.0
      IN = A1 + .5
      RN = IN * 100000
      A1 = CE/100000.0
      IE = A1 + .5
      RE = IE * 100000
      CNP = (CN-RN)*.00001
      CEP = (CE-RE)*.00001
      DO 40 I=1,61
      IF (NR(I).EQ.IN) GO TO 50
   40 CONTINUE
   50 IBLOCK = I
      READ(3'IBLOCK) B1
      IF (IW.EQ.1) GO TO 60
      IF (IW.EQ.0) GO TO 70
```

```
MEMBER NAME  EPASTA
   60 IE = 10 - IE
      GO TO 80
   70 V1   -1.0
      B1(IE,2) = B1(IE,2) * V1
      B1(IE,4)   B1(IE,4) * V1
      B1(IE,6) = B1(IE,6) * V1
      B1(IE,8)   B1(IE,8) * V1
   80 A =(B1(IE,5)+((CNP*B1(IE,7))-(CEP*B1(IE,8))))
      B =(B1(IE,6)+((CNP*B1(IE,8))+(CEP*B1(IE,7))))
      C =(B1(IE,3)+((CNP*A)-(CEP*B)))
      D =(B1(IE,4)+((CNP*B)+(CEP*A)))
      CN =(B1(IE,1)+((CNP*C)-(CEP*D)))
      CE =(500000.0+B1(IE,2)+((CNP*D)+(CEP*C)))
      RETURN
      END
```

```
MEMBER NAME  EPAAREA1
C.... THIS READS IN EPA NEDS AREA SOURCES DATA (TON/YEAR)
C.... IN NON-UNIFORM GRID AND CONVERT THEM INTO UNIFORM
C.... GRID FOR 1-KM GRID.
C.... IXBEG,IYBEG=ORIGIN OF THE GRIDS,IXMAX,IYMAX=AREA SIZE IN KM.
C.... QA=AREA SOURCE STRENTH.
      DIMENSION Q(200,140), IQ(200)
      DATA Q/28000*0.0/,IUNIT/5/,JUNIT/6/,KUNIT/11/,IFORM/25/
     *    ,IXBEG/640/,IYBEG/4190/,IXMAX/200/,IYMAX/140/
      IPT=IXMAX/IFORM
      ICARD=0
      QTOT=0.
C.... START THE LOOP TO READ IN DATA..
1000  CONTINUE
      READ (IUNIT,8010) X,Y,SIZE,QA
8010    FORMAT (2F7.1,F5.1,24X,E12.4)
      IF (X.GE.999.) GO TO 1001
      ICARD=ICARD+1
      IX=X
      IY=Y
      ISIZE=SIZE
      KX=IX-IXBEG+1
      KY=IY-IYBEG+1
      IF (ISIZE.EQ.1) GO TO 110
      KXEND=KX+ISIZE-1
      KYEND=KY+ISIZE-1
      DO 100 I=KX,KXEND
      DO 100 J=KY,KYEND
      IF (Q(I,J).GT.0.0) WRITE (JUNIT,8020) ICARD,I,J
8020  FORMAT (' DUPLICATED SOURCE AT : ICARD,I,J = ',3I5)
      Q(I,J)=QA
C     QTOT=QTOT+QA
100    CONTINUE
      GO TO 120
110   CONTINUE
      IF (Q(KX,KY).GT.0.0) WRITE (JUNIT,8020) ICARD,KX,KY
      Q(KX,KY)=QA
C     QTOT=QTOT+QA
120   CONTINUE
      GO TO 1000
1001  CONTINUE
C.... COMPLETE THE READ LOOP..
C.... SUM THE TOTAL SOURCE QTOT.
      DO 200 J=1,IYMAX
      DO 200 I=1,IXMAX
  200 QTOT=QTOT+Q(I,J)
      WRITE (JUNIT,8030) ICARD,QTOT
8030  FORMAT (' *** NUMBER OF AREA SOURCES = ',I4,' ***'/
     &         ' *** TOTAL SOURCE EMISSION = ',E12.6, ' ***')
C....
C.... PRINT OUT Q ARRAY.
      DO 310 IP=1,IPT
      IA=(IP-1)*IFORM+1
      IB=IA+IFORM-1
      WRITE (JUNIT,8040) IP,(I,I=IA,IB)
      DO 300 JJ=1,IYMAX
      J=(IYMAX+1)-JJ
8040    FORMAT ('1  ***** PAGE = ',I5,/6X,25I5//)
      DO 410 I=IA,IB
```

```
MEMBER NAME  FPAAREA1
  410 IO(I)=Q(I,J)*1000.
      WRITE (JUNIT,8050) J,(IO(I),I=IA,IB)
8050    FORMAT (1X,I4,1X,25I5)
300    CONTINUE
310    CONTINUE
C....
C.... WRITE Q(I,J) ON DISK...
      DO 400 J=1,IYMAX
C     WRITE (KUNIT,8060) (Q(I,J),I=1,IXMAX)
8060   FORMAT (10F7.5,10X)
400    CONTINUE
C     WRITE (KUNIT,8070) QTOT
8070   FORMAT (F12.6)
      RETURN
C     //X.FT11F001 DD DSN=JCS4165.FPAAREA1.DATA,DISP=SHR
C     //X.SYSIN DD DSN=JCS4165.FPAAREA0.DATA,DISP=SHR
      END
```

```
MEMBER NAME  EPASFCZ
      DIMENSION ZSA(30),ZSB(30),ZSC(30),IZSC(30)
      DATA KUNIT/11/,KUNIT2/12/,JUNIT/6/,KUNIT3/13/
C.... READ IN ZSB(TERRAIN HEIGHT) AND ZSA(TOTAL TERRAIN PLUS BUILDING
C.... HEIGHT) AND GET ZSC(BUILDING HEIGHT)
  100 CONTINUE
      READ (KUNIT,8000,END=250)IX,IY,ID,IDX,NMAX,ZSA
 8000 FORMAT(I3,I4,2I1,I2,1X,30F2.0)
      WRITE (JUNIT,8010) IX,IY,ID,IDX,NMAX,(ZSA(I),I=1,NMAX)
      READ(KUNIT2,8000,END=250)IX,IY,ID,IDX,NMAX,ZSB
      DO 200 I=1,NMAX
  200 ZSC(I)=ZSA(I)-ZSB(I)
      WRITE (JUNIT,8020) IX,IY,ID,IDX,NMAX,(ZSB(I),I=1,NMAX)
 8010 FORMAT (/,1X,I4,I5,2I2,I3,21F4.0)
      WRITE (JUNIT,8020) IX,IY,ID,IDX,NMAX,(ZSC(I),I=1,NMAX)
 8020 FORMAT (1X,I4,I5,2I2,I3,21F4.0)
      DO 300 I=1,NMAX
  300 IZSC(I)=ZSC(I)
C     WRITE (KUNIT3,8030)IX,IY,ID,IDX,NMAX,(IZSC(I),I=1,NMAX)
 8030 FORMAT (I3,I4,2I1,I2,1X,21I2)
      GO TO 100
  250 CONTINUE
      STOP
      END
```

```
MEMBER NAME  EPAAREA2
C.... THIS READS IN EPA NEDS AREA SOURCES DATA (TON/YEAR)
C.... IN UNIFORM 1-KM GRID AND CHOOSE A SUB-AREA FOR ANALYSIS.
C.... IXBEG,IYBEG=ORIGIN OF THE GRIDS IN UTM,IXMAX,IYMAX=AREA SIZE IN KM
C.... QA,Q(200,140)=AREA SOURCE STRENTH.
C.... IBEG,JBEG=ORIGIN OF THE SUB-AREA
C.... ISIZE,JSIZE= SIZE OF THE SUB-AREA.
C.... IXUTM,JYUTM=ARRAY OF UTM POSITION.
C.... IA,JA =INDEX ARRAY OF THE NON-UNIFORM GRIDS.
C.... DXB,DYB=GRID SIZE OF THE NON-UNIFORM GRIDS.
C.... ZS(200,140)= TEMP. ARRAY FOR ZSS &ZSBG.
C.... ZSS=AVERAGED TERRAIN HEIGHT(M). ZSBG=BUILDING HEIGHT.
C.... Z0=AREA SOURCE EMISSION HEIGHT OR ROUGHNESS (M).
C....
      DIMENSION ZS(200,140), Q(200,140),IXUTM(200),JYUTM(140),II(200)
     *          ,IA(200),JA(140)
      EQUIVALENCE (ZS(1,1),Q(1,1))
      DATA ZS/28000*-1.0/,IUNIT/5/,JUNIT/6/,KUNIT/11/,IFORM/40/
     *    ,IXBEG/640/,IYBEG/4190/,IXMAX/200/,IYMAX/140/,KUNIT2/12/
     *    ,IBEG/86/,JBEG/63/,ISIZE/40/,JSIZE/60/,DXA/1./,DYA/1./
C.... COMPUTED ARRAYS.
      DIMENSION ZSS(30,40),ZSBG(30,40),Z0(30,40),QQ(30,40)
     *          ,DXB(30),DYB(40)
      DATA IM/30/,JM/40/,DXB/5*2.,20*1.,5*2.0/,DYB/10*2.,20*1.,10*2./
     *    ,RATIO/1.0/,LSUM/0/
      REAL*4 TITLE1(10)/'AREA',' SOU','RCES',' IN ','XXXX',5*'    '/
     *      ,TITLE2(10)/'TERR','AIN ','HEIG','T IN',' M  ',5*'    '/
     *      ,TITLE3(10)/'ROUG','HNES','S IN',' CM ',6*'    '/
     *      ,TITLE4(10)/'AREA',' SOU','RCES',' EMI','SSIO','N HT',
     *                  ' IN ','M   ',2*'    '/
     *      ,TITLE5(10)/'BUIL','DING',' HT ','IN M',6*'    '/
     *      ,FMTDS1(10)/'(10F','7.5,','10X)',7*'    '/
     *      ,FMTDS2(10)/'(18F','4.1,',' 8X)',7*'    '/
     *      ,FMTDS3(10)/'(18F','4.2,',' 9X)',7*'    '/
C....
C.... READ IN Q(200,140) AND CONVERT TO QQ(30,40).
      CALL READQQ(Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXBEG,IYBEG,
     *     IXUTM,JYUTM,JUNIT,KUNIT,SQTOT)
      RATIO=0.0
      CALL WRITEQ(Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM,IFORM
     *        ,JUNIT,II,RATIO,TITLE1,FMTDS1)
      IA(1)=IBEG
      JA(1)=JBEG
      CALL GRIDCN(Q,IXMAX,IYMAX,IA,JA,DXA,DYA,QQ,IM,JM,DXB,DYB,
     *              IXUTM,JYUTM,1,LSUM)
      CALL WRITEQ(QQ,IM,IM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II,RATIO
     *           ,TITLE1,FMTDS1)
C     WRITE (14,FMTDS1) QQ
C     WRITE (14,8000) SQTOT
 8000 FORMAT(F10.5)
C....
C.... READ IN SURFACE HEIGHT ZSA AND EXPAND THEM TO ZS(200,140)
C.... AND PUT IN ZSS(30,40) FOR KK=1: ZSBG FOR KK=2.
      DO 100 KK=1,2
      CALL READZS(ZS,IXMAX,IYMAX,IXBEG,IYBEG,JUNIT,KUNIT2)
C.... PRINT ZS IN THE GIVEN REGION AND COMPUTED REGION.
      DO 200 K=1,2
C.... BUKCVICH'S REGION BEGINS AT (75,66).
      IBEG=75
```

```
MEMBER NAME  EPAAREA2
      JBEG=66
      ISIZE=50
      JSIZE=50
      IF ( K .LT. 2) GO TO 110
      IBEG=86
      JBEG=63
      ISIZE=40
      JSIZE=60
  110 CONTINUE
      DO 300 J=1,JSIZE
  300 JYUTM(J)=IYBEG+JBEG+J-2
      DO 310 I=1,ISIZE
  310 IXUTM(I)=IXBEG+IBEG+I-2
      RATIO=1.0
      CALL WRITEQ(ZS,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM,IFORM
     *       ,JUNIT,II,RATIO,TITLE2,FMTDS2)
  200 CONTINUE
      IA(1)=IBEG
      JA(1)=JBEG
      IF (KK .EQ. 2) GO TO 220
      CALL GRIDCN(ZS,IXMAX,IYMAX,IA,JA,DXA,DYA,ZSS,IM,JM,DXB,DYB,
     *      IXUTM,JYUTM,1,LSUM)
      CALL WRITEQ(ZSS,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II,RATIO
     *          ,TITLE2,FMTDS2)
      GO TO 100
  220 CONTINUE
      CALL GRIDCN(ZS,IXMAX,IYMAX,IA,JA,DXA,DYA,ZSBG,IM,JM,DXB,DYB,
     *           IXUTM,JYUTM,1,LSUM)
      CALL WRITEQ(ZSBG,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II,RATIO
     *          ,TITLE5,FMTDS2)
  100 CONTINUE
C.... CALCULATE EFFECTIVE HEIGHT EMISSION HEIGHT FOR AREA SOURCES.
      DO 400 J=1,JM
      DO 400 I=1,IM
  400 ZO(I,J)=ZSBG(I,J)+10.0
      CALL WRITEQ(ZO,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II,RATIO
     *          ,TITLE4,FMTDS3)
C....
C.... CALCULATE SURFACE ROUGHNESS ZO(30,40) FROM ZSS AND QQ.
C.... ZO=0.5*(ZSS+5)*(0.04*(1.+1.0*QQ/SQTOT)).
      SQTOTI=1.0/SQTOT
      DO 500 J=1,JM
      DO 500 I=1,IM
      ZOZS=0.04*(1.0+1200.0*QQ(I,J)*SQTOTI)
      ZO(I,J)=0.5*ZOZS*(ZSBG(I,J)+0.2*ZSS(I,J)+5.0)
  500 CONTINUE
      RATIO=0.0
      CALL WRITEQ(ZO,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,II,RATIO
     *          ,TITLE3,FMTDS3)
C     WRITE (14,FMTDS3) ZO
      STOP
      END
      SUBROUTINE READZS(ZS,IXMAX,IYMAX,IXBEG,IYBEG,JUNIT,KUNIT)
      DIMENSION ZS(IXMAX,IYMAX),ZSA(30)
C.... THIS READS IN BUKOVICH'S DATA ON TERRAIN HEIGHT(ZSA) AND
C.... BUILDING HEIGHT(ZSC) AND CALCULATE THE EFFECTIVE HT.
C....
      WRITE (JUNIT,8020) KUNIT
```

```
MEMBER NAME  FDAAREA2
 8020 FORMAT('1    *** READ IN BUKOVICH DATA OF TERRAIN AND BUILDING HEIG
     *HT IN HIS GRID REGION FROM UNIT= ',I4)
C.... INITIALIZE ZS=-1.0.
      DO 50 J=1,IYMAX
      DO 50 I=1,IXMAX
   50 ZS(I,J)=-1.0
  100 CONTINUE
      READ (KUNIT,8000,END=250) IX,IY,ID,IDX,NMAX,ZSA
C     READ (13,    8000,END=250) IX,IY,ID,IDX,NMAX,ZSC
 8000 FORMAT(I3,I4,2I1,I2,1X,30F2.0)
      WRITE (JUNIT,8010) IX,IY,ID,IDX,NMAX,(ZSA(I),I=1,NMAX)
C     WRITE (JUNIT,8010) IX,IY,ID,IDX,NMAX,(ZSC(I),I=1,NMAX)
 8010 FORMAT (1X,I4,I5,2I2,I3,21F4.0)
C.... READ ONE CARD AND PUT THEM INTO ZS(KX,KY)
      DO 110 N=1,NMAX
      J1=(ID+1)/2
      I1=ID+1-J1*2
      JX=IX+(N-1)*IDX*I1
      JY=IY+(N-1)*IDX*J1
      KI=JX-IXBEG+1
      KJ=JY-IYBEG+1
      DO 120 J2=1,IDX
      KJJ=KJ-(IDX/2)+J2-1
      DO 120 I2=1,IDX
      KII=KI-(IDX/2)+I2-1
      ZS(KII,KJJ)=ZSA(N)
  120 CONTINUE
  110 CONTINUE
      GO TO 100
  250 CONTINUE
C.... FILL UP THE VOID REGION BY LINEAR INTERPOLATION.
      DO 400 J=76,103,27
      DO 400 I=86,111
  400 ZS(I,J)=0.5*(ZS(I,J-1)+ZS(I,J+1))
      DO 410 I=85,112,27
      DO 410 J=76,103
  410 ZS(I,J)=0.5*(ZS(I-1,J)+ZS(I+1,J))
      C13=1.0/3.0
      C23=2.0/3.0
      DO 420 J=70,108,38
      DO 420 I=81,116
      ZS(I,J)=C23*ZS(I,J-1)+C13*ZS(I,J+2)
  420 ZS(I,J+1)=C23*ZS(I,J+2)+C13*ZS(I,J-1)
      DO 430 I=79,117,38
      DO 430 J=70,109
      ZS(I,J)=C23*ZS(I-1,J)+C13*ZS(I+2,J)
  430 ZS(I+1,J)=C23*ZS(I+2,J)+C13*ZS(I-1,J)
      DO 440 J=80,100
      I=88
      ZS(I,J)=0.5*ZS(I-1,J)+0.5*ZS(I+1,J)
      DO 440 I=89,109
  440 ZS(I,J)=0.5*ZS(I,J)+0.5*ZS(I+1,J)
      DO 450 I=88,109
      J=79
      ZS(I,J)=0.5*ZS(I,J-1)+0.5*ZS(I,J+1)
      DO 450 J=80,100
  450 ZS(I,J)=0.75*ZS(I,J)+0.25*ZS(I,J+1)
      DO 460 J=118,122
```

```
MEMBER NAME  EPAAREA2
      DO 460 I=75,126
  460 ZS(I,J)=ZS(I,117)
      RETURN
      END
      SUBROUTINE WRITEQ (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM
     *                  ,IFORM,JUNIT,II,RATIO,TITLE,FMTDS)
C.... THIS WRITES Q(ISIZE,JSIZE) ON UNIT=JUNIT IN FORM=IFORM.
C.... IFORM=NO. OF COLUMES TO BE PRINTED ON ONE LINE.
C....
      DIMENSION RFMT1(10),RFMT2(10),RFMT3(10),TITLE(10),FMTDS(10)
      LOGICAL*1 FMT1(40),FMT2(40),FMT3(40),FM(8)
      EQUIVALENCE (FM(1),IFM1),(FM(5),IFM2)
     * ,(FMT1(1),RFMT1(1)),(FMT2(1),RFMT2(1)),(FMT3(1),RFMT3(1))
      DATA RFMT1/'(5X,','5HXU','TM=,','40I3',',   )',5*'    '/
     *     ,RFMT2/'(5X,','5H  ',' I=,','40I3',',/ )',5*'    '/
     *     ,RFMT3/'(1X,','2I4,','1X ,','40I3','    )',5*'    '/
      DIMENSION Q(IXMAX,IYMAX),IXUTM(ISIZE),JYUTM(JSIZE) ,II(ISIZE)
     *          ,IQ(200)
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
      IF (IFORM .EQ. 0) GO TO 500
      IFM1=120/IFORM+240
      IFM2=IFORM/10+240
      FMT1(16)=FM(4)
      FMT2(16)=FM(4)
      FMT3(16)=FM(4)
      FMT1(13)=FM(8)
      FMT2(13)=FM(8)
      FMT3(13)=FM(8)
      IPT=(ISIZE-1)/IFORM+1
      IF (RATIO .NE. 0.0) GO TO 100
      RATIO=1000.0
      IF (IFORM .GE. 30) RATIO=100.
C     IF (IFORM .GE. 40) RATIO=10.0
  100 CONTINUE
C.... PRINT THE TITLE SPECIFIED.
      WRITE (JUNIT,8000) TITLE
 8000 FORMAT (1H1,'   *****  TITLE= ',10A4)
      DO 310 IP=1,IPT
      IA=(IP-1)*IFORM+IBEG
      IB=IA+IFORM-1
      DO 320 I=IA,IB
      II(I)=I
      IF((II(I).GT. 100) .AND.(IFORM.GT. 30))II(I)=II(I)-100
  320 CONTINUE
      IAX=IA-IBEG+1
      IBX=IAX+IFORM-1
C     WRITE (JUNIT,8040) IP,(IXUTM(I),I=IAX,IBX),(II(I),I=IA,IB)
C8040 FORMAT('0  ***** PAGE = ',I5,/5X,'IUTM=',40I3,/,5X,'   I=',40I3//)
      WRITE (JUNIT,8040) IP
 8040 FORMAT (/,'  ***** PAGE = ',I5/)
      WRITE (JUNIT,FMT1) (IXUTM(I),I=IAX,IBX)
      WRITE (JUNIT,FMT2) (II(I),I=IA,IB)
      DO 300 JJ=1,JSIZE
      J=(JEND+1)-JJ
      JUTM=JYUTM(JSIZE+1-JJ)
      DO 410 I=IA,IB
  410 IQ(I)=Q(I,J)*RATIO
```

```
MEMBER NAME  EPAAREA2
C      WRITE (JUNIT,8050) JUTM,J,(IQ(I),I=IA,IB)
C8050 FORMAT (1X,I4,I4,1X,40I3)
      WRITE (JUNIT,FMT3) JUTM,J,(IQ(I),I=IA,IB)
  300 CONTINUE
  310 CONTINUE
      RETURN
  500 CONTINUE
C.... THIS WRITES Q ON DISK IN CARD IMAGE USING FORMAT=FMTDS....
C....
C.... WRITE Q(I,J) ON DISK...
C      DO 400 J=1,IYMAX
C      WRITE (JUNIT,8060) (Q(I,J),I=1,IXMAX)
C8060 FORMAT (10F7.5,10X)
C      WRITE (JUNIT,FMTDS) (Q(I,J),I=1,IXMAX)
  400 CONTINUE
      WRITE (JUNIT,FMTDS) Q
      WRITE (6,8010) TITLE
 8010 FORMAT(/,'  ****** TITLE =',10A4,' HAS BEEN WRITTEN ON DISK')
      RETURN
      END
      SUBROUTINE GRIDCN(Q,IN,JN,IA,JA,DXA,DYA,QQ,IM,JM,DXB,DYB,
     *                  IXUTM,JYUTM,L,LSUM)
C....
C.... L=1,THIS CONVERTS Q(IN,JN) IN UNIFORM GRID INTO QQ(IM,JM) IN NON
C.... UNIFORM GRID.
C.... L=2, CONVERT QQ INTO Q.
C.... LSUM=1,TAKE SUM; LSUM=0,TAKE AVERAGE.
C.... IA,JA=LOCATION IN QQ GRID WHERE GRID SPACECHANGES, EXCEPT
C.... IA(1),JA(1) DENOTES THE ORIGIN OF QQ---QQ(1,1)=Q(IA(1),JA(1)).
C.... DX,DY=GRID SPACE WITH INITIAL POINT AT (IA,JA).
C....
      DIMENSION Q(IN,JN),QQ(IM,JM),IA(IM),JA(JM),DXB(IM),DYB(JM)
     *         ,IXUTM(IM),JYUTM(JM)
C....
      DXAI=1.0/DXA
      DYAI=1.0/DYA
      AREAI=DXAI*DYAI
      IBEG=IA(1)
      JBEG=JA(1)
      IXBEG=IXUTM(1)
      IYBEG=JYUTM(1)
C.... SET UP THE TABLE FOR INDICIES OF Q(I,J).
C....
      JM1=JM-1
      IM1=IM-1
      DO 20 J=1,JM1
      JINTVL=DYB(J)*DYAI+0.2
      JA(J+1)=JA(J)+JINTVL
      JYUTM(J+1)=JYUTM(J)+JINTVL
   20 CONTINUE
      DO 40 I=1,IM1
      IINTVL=DXB(I)*DXAI+0.2
      IA(I+1)=IA(I)+IINTVL
      IXUTM(I+1)=IXUTM(I)+IINTVL
   40 CONTINUE
C      DEBUG INIT(IA,JA)
C.... TRANSFORM Q(KI,KJ) TO QQ(I,J).
      DO 600 J=1,JM
```

```
MEMBER NAME  EPAAREA2
      KJ=JA(J)
      JJJEND=DYB(J)*DYAI+0.2
      DO 500 I=1,IM
      KI=IA(I)
      IIIEND=DXB(I)*DXAI+0.2
C....
C.... AVERAGE THE Q(KI,KJ) TO GET QQ(II,JJ)
      IF (LSUM .EQ. 0) AREAI=1.0/(DXB(I)*DYB(J))
      QTOT=0.0
      DO 200 JJJ=1,JJJEND
      KJJ=KJ+JJJ-1
C....
      DO 100 III=1,IIIEND
      KII=KI+III-1
      QTOT=QTOT+Q(KII,KJJ)
  100 CONTINUE
  200 CONTINUE
      QQ(I,J)=QTOT*AREAI
  300 CONTINUE
  400 CONTINUE
  500 CONTINUE
  600 CONTINUE
      RETURN
C     DEBUG INIT (IA,JA,KI,KJ)
      END
      SUBROUTINE READQQ(Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXBEG,IYBEG,
     *          IXUTM,JYUTM,JUNIT,KUNIT,SQTOT)
C.... THIS READS IN EPA NEDS AREA SOURCES DATA (TON/YEAR)
C.... IN UNIFORM 1-KM GRID AND CHOOSE A SUB-AREA FOR ANALYSIS.
C.... IXBEG,IYBEG=ORIGIN OF THE GRIDS IN UTM,IXMAX,IYMAX=AREA SIZE IN KM
C.... QA,Q(200,140)=AREA SOURCE STRENTH.
C.... IBEG,JBEG=ORIGIN OF THE SUB-AREA
C.... ISIZE,JSIZE= SIZE OF THE SUB-AREA.
C....
      DIMENSION Q(IXMAX,IYMAX), IXUTM(IXMAX),JYUTM(IYMAX)
C....
C.... READ IN Q(200,140) AND QTOT.
      REWIND KUNIT
      DO 100 J=1,IYMAX
      READ (KUNIT,8060) (Q(I,J),I=1,IXMAX)
 8060 FORMAT (10F7.5,10X)
  100 CONTINUE
      READ (KUNIT,8070) QTOT
 8070 FORMAT (F12.6)
C....
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
      ITOT=ISIZE*JSIZE
C.... SUM THE TOTAL SOURCE QTOT FOR SUB-AREA.
      SQTOT=0.0
      DO 200 J=JBEG,JEND
      DO 200 I=IBEG,IEND
  200 SQTOT=SQTOT+Q(I,J)
      RQTOT=SQTOT/QTOT
      WRITE (JUNIT,8030) ITOT,SQTOT,QTOT,RQTOT
 8030 FORMAT (' *** NUMBER OF AREA SOURCES = ',I10,' ***'/
     &        ' *** TOTAL SOURCE EMISSION & RATIO= ',3G13.5, ' ***')
C....
```

```
MEMBER NAME  EPAAREA2
      DO 300 J=1,JSIZE
  300 JYUTM(J)=IYBEG+JBEG+J-2
      DO 310 I=1,ISIZE
  310 IXUTM(I)=IXBEG+IBEG+I-2
      RETURN
      END
```

```
MEMBER NAME  EPAGE01
C.... THIS PROGRAM CREATES GEOGRAPHY AND ANNUAL EMISSION DATA
C       ON UNIT=KUNIT, DSN=EPAGE01.DATA
      DIMENSION XP(200),YP(200),ZP(200),QP(200),ZR(200),NN(200)
      DIMENSION FMTDS(10)
      REAL*4 FMTDS1(10)/'(3(I','5,F6','.1,F','7.1,','F6.1','))  '
     &                  ,4*'    '/
     &      ,FMTDS2(10)/'(9F8','.2) ',8*'    '/
     &      ,FMTDS3(10)/'(12F','6.1)',8*'    '/
      LOGICAL*1 A(80)
      DATA IXBEG/640/,IYBEG/4190/,IBEG/86/,JBEG/63/
      DATA IUNIT1/11/,IUNIT2/12/,IUNIT3/13/,IUNIT4/14/
      DATA IUNIT/10/,JUNIT/6/,KUNIT/15/
      WRITE (KUNIT,8010) IXBEG,IYBEG,IBEG,JBEG
8010  FORMAT (4I5)
10    READ (IUNIT,8000,END=100) (A(I),I=1,80)
      WRITE (KUNIT,8000) (A(I),I=1,80)
      GO TO 10
100   READ (IUNIT1,8000,END=200) (A(I),I=1,80)
      WRITE (KUNIT,8000) (A(I),I=1,80)
8000  FORMAT (80A1)
      GO TO 100
200   READ (IUNIT2,8000,END=300) (A(I),I=1,80)
      WRITE (KUNIT,8000) (A(I),I=1,80)
      GO TO 200
300   CONTINUE
      READ (IUNIT3,8300) FMTDS
8300  FORMAT (10A4)
      N=1
      QPTOT=0.
302   READ (IUNIT3,FMTDS,END=400) NN(N),XP(N),YP(N),ZP(N),QP(N),ZR(N)
      QP(N)=QP(N)*0.02876664
      QPTOT=QPTOT+QP(N)
      N=N+1
      GO TO 302
400   CONTINUE
      NMAX=N-1
      WRITE (KUNIT,8310) NMAX
      WRITE (KUNIT,8300) FMTDS1
      WRITE (KUNIT,FMTDS1) (NN(K),XP(K),YP(K),ZP(K),K=1,NMAX)
8310  FORMAT (I5)
500   READ (IUNIT4,8000,END=600) (A(I),I=1,80)
      WRITE (KUNIT,8000) (A(I),I=1,80)
      GO TO 500
600   CONTINUE
      WRITE (KUNIT,8300) FMTDS2
      WRITE (KUNIT,FMTDS2) (QP(K),K=1,NMAX)
      WRITE (KUNIT,8620) QPTOT
      WRITE (KUNIT,8300) FMTDS3
      WRITE (KUNIT,FMTDS3) (ZR(K),K=1,NMAX)
8620  FORMAT (F10.1)
C.... PRINT DATA SET ON UNIT=KUNIT.
      REWIND KUNIT
      WRITE (JUNIT,8630)
8630  FORMAT ('1 ***** DATA ON UNIT=KUNIT *****'///)
700   READ (KUNIT,8000,END=800) (A(I),I=1,80)
      WRITE (JUNIT,8700) (A(I),I=1,80)
8700  FORMAT (5X,80A1)
      GO TO 700
```

```
MEMBER NAME  FPAGE01
800    CONTINUE
       STOP
       END
```

MEMBER NAME EPAGEOIN

```
C
      DIMENSION QB(30,40),ZS(30,40),Z0(30,40)
     &      ,PQB(150),XP(150),YP(150),ZP(150),ZR(150),NP(150)
     &      ,XRAMS(25),YRAMS(25),IRAMS(25)
     &      ,IXUTM(30),JYUTM(40),DXB(30),DYB(40)
      DATA DXB/5*2.,20*1.,5*2./,DYB/10*2.,20*1.,10*2./
      DATA IM/30/,JM/40/,LM/150/,NS/25/
      CALL GEOIN (QB,ZS,PQB,XP,YP,ZP,ZR,Z0,NP,XRAMS,YRAMS,IRAMS
     &      ,IXUTM,JYUTM,DXB,DYB,IM, JM,LM,LMM,NS,NRAMS)
      NS=NRAMS
      LM=LMM
      STOP
      END
       SUBROUTINE GEOIN (QB,ZS,PQB,XP,YP,ZP,ZR,Z0,NP,XRAMS,YRAMS,IRAMS
     &                   ,IXUTM,JYUTM,DXB,DYB,IM,JM,LM,LMMAX,NS,NRAMS)
C.... THIS ROUTINE READ IN GEOGRAPHICAL AND ANNUAL EMISSION DATA.
C     IT ALSO: 1) FIX UTM COORDINATES FOR ALL NUMERICAL GRIDS.
C              2) PRINT DATA SETS
C              3) CONVERT LOCATION OF POINT SOURCE TO NUMERICAL GRID.
      DIMENSION QB(IM,JM),ZS(IM,JM),Z0(IM,JM)
     &      ,PQB(LM),XP(LM),YP(LM),ZP(LM),ZR(LM),NP(LM)
     &      ,XRAMS(NS),YRAMS(NS),IRAMS(NS)
     &      ,IXUTM(IM),JYUTM(JM),DXB(IM),DYB(JM)
       DIMENSION FMTDS(10)
      DATA JUNIT/6/,KUNITG/15/
      DATA RATIO/1.0/,DXA/1.0/,DYA/1.0/
      REAL*4 TITLE1(10)/'AREA',' SOU','RCES',' IN ','????',5*'    '/
     *      ,TITLE2(10)/'ROUG','HNES','S IN',' M  ',6*'    '/
     *      ,TITLE3(10)/'AREA',' SOU','RCES',' EMI','SSIO','N HT',
     *                  ' IN ',' M  ',2*'    '/
C.... READ ORGIN OF UMT COORDINATE AND FIX FOR ALL GRIDS.
      READ (KUNITG,8000) IXBEG,IYBEG,IBEG,JBEG
8000  FORMAT (4I5)
      CALL XYUTMS (IXUTM,JYUTM,IXBEG,IYBEG,IBEG,JBEG,DXB,DYB,DXA,DYA
     &              ,IM,JM)
C.... READ IN UTM COORDINATES OF RAMS STATIONS
      READ (KUNITG,8030) NRAMS
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) (IRAMS(L),XRAMS(L),YRAMS(L),L=1,NRAMS)
C.... PRINT AND CONVERT RAMS LOCATION TO NUMERICAL GRID
      WRITE (JUNIT,8015) NRAMS
8015  FORMAT ('1 ***** LOCATION OF ',I2,' RAMS STATIONS *****'//
     &      '    IS',4X,'XRAMS',5X,'YRAMS',6X,'XS',6X,'YS')
      DO 100 L=1,NRAMS
      XI=XRAMS(L)
      YJ=YRAMS(L)
      CALL XYUTM1 (XI,YJ)
      WRITE (JUNIT,8020) IRAMS(L),XRAMS(L),YRAMS(L),XI,YJ
8020  FORMAT (I5,2F10.2,2F8.2)
      XRAMS(L)=XI
      YRAMS(L)=YJ
100   CONTINUE
C.... READ IN SURFACE ROUGHNESS AND PRINT
      READ (KUNITG,8010) FMTDS
8010  FORMAT (10A4)
      READ (KUNITG,FMTDS) Z0
      RATIO=0.0
      CALL WRITEQ (Z0,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,RATIO,TITLE2)
```

```
MEMBER NAME  FPAGEOIN
C.... READ IN AREA SOURCE EMISSION HEIGHT AND PRINT.
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) ZS
      RATIO=1.0
      CALL WRITEO (ZS,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,RATIO,TITLE3)
C.... READ IN POINT SOURCES LOCATION AND STACK HEIGHT.
      READ (KUNITG,8030) LMMAX
8030  FORMAT (I5)
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) (NP(L),XP(L),YP(L),ZP(L),L=1,LMMAX)
C.... READ IN AREA SOURCE EMISSION RATES AND PRINT.
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) QB
      READ (KUNITG,8050) QBTOT
8050  FORMAT (F10.4)
      RATIO=0.0
      CALL WRITEO (QB,IM,JM,IM,JM,1,1,IXUTM,JYUTM,IM,JUNIT,RATIO,TITLE1)
      WRITE (JUNIT,8070) QBTOT
8070  FORMAT (//5X,'.... TOTAL EMISSION = ',F10.2)
C.... READ IN POINT SOURCE EMISSION RATES.
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) (PQB(L),L=1,LMMAX)
      READ (KUNITG,8090) PQBTOT
8090  FORMAT (F10.1)
C.... READ IN PLUME RISE DATA.
      READ (KUNITG,8010) FMTDS
      READ (KUNITG,FMTDS) (ZR(L),L=1,LMMAX)
C.... PRINT POINT SOURCE DATA.
      WRITE (JUNIT,8200) LMMAX
8200  FORMAT ('1 *** POINT SOURCE DATA ***'/
     &        5X,'..... TOTAL NUMBER =',I5//
     &        '    ID',8X,'XP',8X,'YP',8X,'ZP',8X,'QP',8X,'ZR',6X,'XI'
     &           ,6X,'YJ')
C.... CONVERT XP,YP FROM UTM COORDINATE TO NUMERICAL GRID
      DO 200 L=1,LMMAX
      XI=XP(L)
      YJ=YP(L)
      CALL XYUTM1 (XI,YJ)
      WRITE (JUNIT,8210) NP(L),XP(L),YP(L),ZP(L),PQB(L),ZR(L),XI,YJ
8210  FORMAT (I5,3F10.1,F10.2,F10.1,2F8.2)
      XP(L)=XI
      YP(L)=YJ
200   CONTINUE
      WRITE (JUNIT,8070) PQBTOT
      RETURN
      END
      SUBROUTINE XYUTMS (IXUTM,JYUTM,IXBEG,IYBEG,IBEG,JBEG,DXB,DYB
     &                  ,DXA,DYA,IM,JM)
C.... THIS ROUTINE COMPUTES UTM COORDINATES FOR ALL NUMERICAL GRIDS.
C.... ENTRY XYUTM1 CONVERT (X,Y) FROM UTM COORDINATES TO NUMERICAL GRID.
      DIMENSION IXUTM(IM),JYUTM(JM),DXB(IM),DYB(JM)
      IM1=IM-1
      JM1=JM-1
      DXAI=1.0/DXA
      DYAI=1.0/DYA
      IXUTM(1)=IXBEG+IBEG-1
      JYUTM(1)=IYBEG+JBEG-1
      DO 20 J=1,JM1
```

```
MEMBER NAME  EPAGEOIN
      JINTVL=DYB(J)*DYAI+0.2
      JYUTM(J+1)=JYUTM(J)+JINTVL
20    CONTINUE
      DO 40 I=1,IM1
      IINTVL=DXB(I)*DXAI+0.2
      IXUTM(I+1)=IXUTM(I)+IINTVL
40    CONTINUE
      RETURN
C
      ENTRY XYUTM1 (XI,YJ)
C
      LXI=XI
      LYJ=YJ
      IF ((LXI.LT.IXUTM(1)).OR.(LXI.GT.(IXUTM(IM)+DXB(IM)))) GO TO 90
      IF ((LYJ.LT.JYUTM(1)).OR.(LYJ.GT.(JYUTM(JM)+DYB(JM)))) GO TO 90
      DO 60 J=1,JM
      IF (LYJ.GE.JYUTM(J)) GO TO 60
      YJJ=J-1
      DY=(YJ-JYUTM(J-1))/DYB(J-1)
      YJ=YJJ+DY
      GO TO 62
60    CONTINUE
62    CONTINUE
      DO 80 I=1,IM
      IF (LXI.GE.IXUTM(I)) GO TO 80
      XII=I-1
      DX=(XI-IXUTM(I-1))/DXB(I-1)
      XI=XII+DX
      GO TO 82
80    CONTINUE
82    CONTINUE
      RETURN
90    CONTINUE
      XI=-1.0
      YJ=-1.0
      RETURN
      END
      SUBROUTINE WRITEQ (Q,IXMAX,IYMAX,ISIZE,JSIZE,IBEG,JBEG,IXUTM,JYUTM
     *                   ,IFORM,JUNIT,RATIO,TITLE)
C.... THIS WRITES Q(ISIZE,JSIZE) ON UNIT=JUNIT IN FORM=IFORM.
C.... IFORM=NO. OF COLUMES TO BE PRINTED ON ONE LINE.
C....
      DIMENSION RFMT1(10),RFMT2(10),RFMT3(10)
      DIMENSION TITLE(10),FMTDS(10)
      LOGICAL*1 FMT1(40),FMT2(40),FMT3(40),FM(8)
      EQUIVALENCE (FM(1),IFM1),(FM(5),IFM2)
     * ,(FMT1(1),RFMT1(1)),(FMT2(1),RFMT2(1)),(FMT3(1),RFMT3(1))
      DATA RFMT1/'(5X,','5HXU','TM=,','40I3',' )',5*' '/
     *     ,RFMT2/'(5X,','5H ',' I=,','40I3',',/ )',5*' '/
     *     ,RFMT3/'(1X,','2I4,','1X ,','40I3',' )',5*' '/
      DIMENSION Q(IXMAX,IYMAX),IXUTM(ISIZE),JYUTM(JSIZE)
     *          ,IQ(200),II(200)
      JEND=JBEG+JSIZE-1
      IEND=IBEG+ISIZE-1
      IF (IFORM .EQ. 0) GO TO 500
      IFM1=120/IFORM+240
      IFM2=IFORM/10+240
      FMT1(16)=FM(4)
```

```
MEMBER NAME  FPAGEOIN
       FMT2(16)=FM(4)
       FMT3(16)=FM(4)
       FMT1(13)=FM(8)
       FMT2(13)=FM(8)
       FMT3(13)=FM(8)
       IPT=(ISIZE-1)/IFORM+1
       IF (RATIO.NE.0.0) GO TO 100
       RATIO=1000.0
       IF (IFORM .GE. 30) RATIO=100.
 100    CONTINUE
C.... PRINT THE SPECIFIED TITLE
       WRITE (JUNIT,8000) TITLE
8000   FORMAT ('1    ***** TITLE= ',10A4)
       DO 310 IP=1,IPT
       IA=(IP-1)*IFORM+IBEG
       IB=IA+IFORM-1
       DO 320 I=IA,IB
       II(I)=I
       IF((II(I).GT. 100) .AND.(IFORM.GT. 30))II(I)=II(I)-100
  320  CONTINUE
       IAX=IA-IBEG+1
       IBX=IAX+IFORM-1
C      WRITE (JUNIT,8040) IP,(IXUTM(I),I=IAX,IBX),(II(I),I=IA,IB)
C8040  FORMAT('1   ***** PAGE = ',I5,/5X,'IUTM=',40I3,/,5X,'    I=',40I3//)
       WRITE (JUNIT,8040) IP
 8040  FORMAT ('    ***** PAGE    ',I5/)
       WRITE (JUNIT,FMT1) (IXUTM(I),I=IAX,IBX)
       WRITE (JUNIT,FMT2) (II(I),I=IA,IB)
       DO 300 JJ=1,JSIZE
       J=(JEND+1)-JJ
       JUTM=JYUTM(JSIZE+1-JJ)
       DO 410 I=IA,IB
  410  IQ(I)=Q(I,J)*RATIO
C      WRITE (JUNIT,8050) JUTM,J,(IQ(I),I=IA,IB)
C8050  FORMAT (1X,I4,I4,1X,40I3)
       WRITE (JUNIT,FMT3) JUTM,J,(IQ(I),I=IA,IB)
  300  CONTINUE
  310  CONTINUE
       RETURN
  500  CONTINUE
C.... THIS WRITES Q ON DISK IN CARD IMAGE.
C....
C.... WRITE Q(I,J) ON DISK...
C      DO 400 J=1,IYMAX
C      WRITE (KUNIT,8060) (Q(I,J),I=1,IXMAX)
C8060  FORMAT (10F7.5,10X)
  400  CONTINUE
C      WRITE (KUNIT,8070) CTOT
C8070  FORMAT (E12.6)
       RETURN
       END
```

```
MEMBER NAME  EPAMAP
      LOGICAL*1 M(150,200),N(150,200),NO,YES
      DIMENSION MX(10)
      DATA NO/' '/,YES/'1'/
      DATA JUNIT/6/,KUNIT/11/,KUNIT1/12/
      DO 801 I=1,150
      DO 801 J=1,200
      N(I,J)=NO
801   M(I,J)=NO
      DO 200 J=1,200
      READ (KUNIT,850) JJ,(M(I,J),I=1,75)
  850 FORMAT (I3,2X,75A1)
      READ (KUNIT,850) JJ,(M(I,J),I=76,150)
  200 CONTINUE
      ISHIFT=26
      IIMAX=150-ISHIFT
      DO 300 J=1,200
      JJ=J
      DO 300 I=1,IIMAX
      II=I+ISHIFT
      N(I,J)=M(II,JJ)
  300 CONTINUE
      DO 701 K=1,2
      WRITE (JUNIT,491)
491   FORMAT (1H1)
      I1=(K-1)*75+1
      I2=(K-1)*75+75
      DO 702 JJ=1,200
      J=201-JJ
      WRITE (JUNIT,492) (M(I,J),I=I1,I2)
492   FORMAT(5X,75A1)
702   CONTINUE
701   CONTINUE
      DO 600 K=1,5
      I1=(K-1)*30+1
      I2=(K-1)*30+30
      WRITE (JUNIT,691) (I,I=I1,I2)
691   FORMAT (1H1,4X,30(1X,I3))
      WRITE (JUNIT,692)
      DO 601 JJ=1,200
      J=201-JJ
692   FORMAT (6X,119('_'))
      WRITE (JUNIT,693) J,(M(I,J),I=I1,I2)
693   FORMAT (2X,I3,30('|',2X,A1),'|'/5X,30('|___'),'|')
601   CONTINUE
      WRITE (JUNIT,691) (I,I=I1,I2)
600   CONTINUE
      DO 650 J=1,200
C     WRITE (KUNIT1,891) J,(M(I,J),I=1,75)
891   FORMAT (I3,',1',75A1)
C     WRITE (KUNIT1,892) J,(M(I,J),I=76,150)
892   FORMAT (I3,',2',75A1)
650   CONTINUE
      STOP
      END
```

3. Input Data Listing - Geographical and Annual Emission Data Set

```
MEMBER NAME   EPAGED2
  640 4190    86   63
   25
(3(I5,F9.3,F10.3))
    1  744.183  4279.862    2  742.518  4286.044    3  747.588  4282.466
    4  747.312  4277.304    5  743.706  4276.452    6  738.660  4277.565
    7  740.179  4282.610    8  748.407  4291.102    9  755.802  4279.886
   10  747.208  4272.826   11  738.812  4272.478   12  733.938  4280.912
   13  737.738  4289.819   14  744.320  4297.456   15  757.111  4297.799
   16  762.777  4290.082   17  760.560  4272.818   18  743.065  4263.256
   19  729.759  4270.547   20  723.079  4285.908   21  732.414  4302.375
   22  741.631  4329.222   23  777.320  4286.378   24  749.275  4236.537
   25  697.445  4282.239
(18F4.2,8X)
0.480.320.330.330.260.190.190.190.190.190.190.190.290.290.290.290.290.29
0.290.290.280.280.280.280.280.280.280.270.270.270.490.320.330.330.270.19
0.190.190.190.190.190.190.290.290.290.280.280.280.280.280.290.290.290.29
0.290.290.280.280.280.280.490.330.330.330.270.200.200.200.200.200.190.19
0.280.280.280.280.280.280.280.280.300.300.300.300.300.300.290.290.290.29
0.490.350.360.370.310.240.200.200.200.200.250.250.270.440.640.290.290.29
0.300.300.310.310.300.300.300.300.320.320.330.330.480.470.480.450.430.46
0.270.250.230.230.230.230.240.240.300.310.310.310.360.360.370.370.310.31
0.310.320.350.350.340.340.460.530.540.490.480.530.300.270.250.250.230.23
0.240.250.330.320.320.320.380.380.380.380.320.320.320.320.350.350.340.34
0.460.550.580.540.520.580.320.290.260.240.210.240.240.350.420.330.330.30
0.340.340.340.340.310.320.320.310.340.350.650.520.480.570.600.580.550.50
0.560.280.290.160.150.260.260.660.660.380.380.240.240.240.240.260.310.32
0.320.300.350.471.191.280.520.540.500.450.420.350.440.420.830.220.170.22
0.250.570.590.320.310.220.210.230.230.270.320.340.340.370.441.020.970.81
0.560.550.510.700.420.370.571.531.031.380.360.160.200.360.350.430.450.40
0.400.410.360.420.380.380.380.440.920.740.470.530.570.590.600.450.420.39
0.581.621.911.751.470.190.120.270.410.530.450.300.320.390.361.040.380.38
0.380.520.670.420.460.490.790.980.920.840.750.391.182.131.892.612.590.71
0.210.220.330.470.310.200.270.330.700.550.260.260.260.700.490.420.450.44
1.121.091.200.660.612.453.792.211.862.773.112.990.990.280.360.500.370.28
0.320.360.760.440.260.310.260.380.360.780.450.440.810.930.890.902.394.57
4.421.551.784.492.932.792.560.650.360.340.400.300.301.060.540.450.810.26
0.800.860.380.450.450.390.530.630.490.410.580.962.871.401.441.164.074.09
2.511.810.390.320.661.341.150.980.320.290.660.880.980.370.360.450.450.39
0.710.470.570.681.170.881.811.511.744.172.393.312.073.310.570.320.481.72
2.191.111.300.960.360.240.240.250.490.340.370.350.500.750.610.600.992.23
1.750.762.591.621.231.900.910.630.910.380.400.951.611.660.551.040.360.24
0.480.240.280.290.300.310.570.520.510.701.010.450.450.451.241.440.712.12
1.002.251.330.430.450.401.411.450.901.190.770.600.240.240.680.290.300.31
0.540.530.530.590.590.720.430.802.673.492.073.424.902.770.680.300.380.43
0.380.340.960.710.770.810.240.290.300.350.410.300.550.580.520.680.622.27
5.023.292.403.364.203.383.393.311.290.290.540.440.370.340.290.500.330.17
0.170.230.290.350.410.300.660.720.760.720.792.042.433.703.183.762.733.65
3.052.310.790.290.370.410.320.240.210.200.180.310.170.230.290.360.410.30
0.610.810.690.700.690.724.173.863.972.942.912.643.041.170.540.320.690.56
0.240.210.210.200.180.170.170.240.360.681.000.360.600.740.710.690.580.61
0.622.981.652.302.693.152.030.580.400.400.940.750.200.150.160.170.180.17
0.300.240.360.660.630.420.640.910.800.650.640.810.951.141.092.331.260.45
0.510.160.260.371.171.290.440.150.150.150.150.150.150.190.320.480.870.71
0.720.880.790.830.921.001.000.852.281.540.960.650.370.130.230.490.370.77
0.350.240.240.190.150.330.150.190.320.390.740.490.720.970.790.790.690.74
0.810.752.100.860.830.620.250.110.160.280.880.841.311.040.700.190.150.15
0.150.190.320.350.350.530.611.010.900.790.760.770.730.630.651.711.730.37
0.240.110.110.230.340.801.150.841.480.470.150.150.150.190.320.350.350.38
```

```
MEMBER NAME  EPAGEO2
0.710.770.860.790.750.730.610.580.940.570.460.350.310.130.130.170.250.25
0.500.690.770.600.170.250.170.200.320.350.350.630.710.780.490.650.580.53
0.450.680.530.480.490.480.510.190.190.150.150.170.180.220.860.390.730.17
0.590.220.340.380.380.380.530.820.990.730.630.850.650.600.720.660.510.43
0.510.260.230.150.150.150.170.220.430.570.170.370.200.210.330.370.370.37
0.490.661.170.640.610.550.540.660.680.550.530.480.530.500.250.200.200.20
0.150.170.170.190.160.240.160.200.330.370.370.370.400.440.430.600.680.51
0.410.570.590.410.410.410.400.390.220.220.220.220.160.160.290.150.150.22
0.150.190.320.360.370.360.360.440.430.620.710.550.380.530.380.500.410.41
0.380.360.220.210.210.210.160.160.170.170.170.170.170.200.320.360.360.39
0.230.360.450.460.390.410.270.270.310.310.280.280.190.190.160.170.170.17
0.160.160.220.220.220.220.220.230.280.300.300.480.220.280.270.270.260.25
0.250.250.250.250.240.240.150.150.150.150.150.150.150.150.240.240.350.23
0.250.330.230.240.300.460.220.280.270.270.260.250.250.250.250.250.240.24
0.150.150.150.150.150.150.150.150.250.250.340.290.380.230.230.240.240.24
0.220.280.270.270.260.250.250.250.250.250.240.240.150.150.150.150.150.16
0.150.150.230.410.440.710.320.290.230.240.240.240.210.270.280.280.260.25
0.250.250.250.250.240.240.150.160.150.240.170.170.150.160.690.770.640.23
0.460.240.230.240.250.250.210.270.280.280.260.250.250.250.250.250.250.25
0.160.470.350.630.250.540.520.300.230.230.240.230.550.280.270.310.250.25
0.230.290.310.310.250.260.260.260.260.260.320.320.230.350.340.190.170.29
0.320.210.250.250.290.340.340.570.240.240.240.24
(18F4.1,8X)
19.018.018.018.015.012.012.012.012.012.012.012.010.010.010.010.010.010.0
10.010.010.010.010.010.010.010.010.010.010.010.019.018.018.018.015.012.0
12.012.012.012.012.012.010.010.010.010.010.010.010.010.010.010.010.010.0
10.010.010.010.010.010.019.018.018.018.015.012.012.012.012.012.012.012.0
10.010.010.010.010.010.010.010.010.010.010.010.010.010.010.010.010.010.0
19.218.418.518.415.813.312.512.512.512.511.811.810.210.210.210.210.210.2
10.210.210.210.210.210.210.210.511.712.012.012.019.820.120.520.119.218.7
14.514.514.514.511.211.210.810.810.810.810.810.810.810.810.810.810.810.8
10.811.413.514.014.014.020.020.521.020.520.020.015.015.015.015.011.011.0
11.011.011.011.011.011.011.011.011.011.011.011.011.011.513.514.014.014.0
20.020.420.820.420.019.315.515.015.014.011.011.811.813.013.012.012.010.8
10.810.810.811.111.511.511.511.913.614.014.014.020.020.020.020.020.017.0
17.015.015.011.011.014.014.019.019.015.015.010.010.010.010.011.012.012.0
12.012.313.714.014.014.020.020.020.020.020.019.319.318.416.513.012.213.3
14.118.118.114.814.311.611.112.012.012.813.513.513.513.613.914.014.014.0
20.020.120.120.020.020.020.020.019.818.614.012.113.815.414.814.314.813.9
14.314.014.014.515.015.015.014.814.214.014.014.020.020.420.420.020.020.0
20.020.020.020.416.611.310.112.817.019.116.612.813.916.016.015.515.015.0
15.014.814.214.014.014.020.020.020.020.020.020.020.020.120.320.120.116.4
12.014.519.419.914.411.014.116.016.016.016.016.015.714.314.014.014.0
20.020.020.020.020.020.020.120.821.020.420.420.517.617.320.821.016.413.9
16.518.018.017.016.016.016.015.714.314.014.014.020.020.120.020.020.120.0
20.621.520.920.020.020.420.917.017.320.919.418.518.418.018.017.016.016.0
16.015.211.811.011.011.020.020.420.220.420.620.120.921.120.420.020.020.0
20.519.318.320.020.520.019.519.019.017.516.016.016.015.211.811.011.011.0
20.020.020.821.320.620.621.020.820.620.420.320.320.821.819.118.520.520.0
19.519.019.017.015.015.015.014.311.711.011.011.020.020.020.820.420.020.9
22.322.822.421.821.521.521.922.619.518.620.820.119.519.019.017.015.015.0
15.014.311.711.011.011.020.020.019.620.320.020.421.121.621.822.122.622.5
22.522.819.419.021.820.919.519.019.017.015.015.015.014.311.711.011.011.0
20.020.018.620.319.819.620.421.421.420.821.120.820.821.618.818.922.522.1
19.618.018.016.515.015.015.014.311.711.011.011.020.020.120.820.419.418.9
21.123.021.920.020.020.020.020.618.318.422.322.419.418.018.015.012.012.0
12.011.811.211.011.011.021.020.320.020.020.020.020.020.020.020.020.020.0
20.120.918.818.321.521.417.114.014.013.012.012.012.011.811.211.011.011.0
```

```
MEMBER NAME  EPAGEC2
21.020.420.220.020.020.020.020.020.120.120.020.021.320.818.319.621.419.3
15.414.014.013.012.012.012.012.514.515.015.015.020.020.820.920.120.020.1
20.420.520.820.420.120.321.619.118.521.520.918.313.411.011.011.512.012.0
12.012.514.515.015.015.020.020.821.120.220.020.421.321.921.420.120.621.3
16.411.816.921.820.817.813.611.011.011.011.011.011.011.714.315.015.015.0
20.020.020.420.019.919.920.421.321.020.520.921.116.410.815.621.521.318.8
15.915.015.013.011.011.011.011.714.315.015.015.020.020.020.020.019.819.6
20.020.420.820.420.518.513.010.012.317.820.619.817.515.015.013.011.011.0
11.011.714.315.015.015.020.020.020.120.120.020.020.020.020.120.120.618.6
13.010.010.014.920.119.618.418.018.014.511.011.011.011.714.315.015.015.0
20.019.018.919.719.520.020.020.020.420.621.019.314.810.810.812.515.915.5
15.118.018.015.012.012.012.012.514.515.015.015.020.016.015.018.018.020.0
20.020.020.021.021.020.020.013.013.011.011.011.011.014.014.013.012.012.0
12.012.514.515.015.015.020.016.015.018.018.020.020.020.020.021.021.020.0
20.013.013.011.011.011.011.014.014.013.012.012.012.012.715.316.016.016.0
16.316.016.518.018.819.318.518.518.518.818.017.817.816.013.012.512.512.5
11.011.811.811.611.511.511.512.215.216.016.016.015.016.017.018.019.019.0
18.018.018.018.017.017.017.017.013.013.013.013.011.011.011.011.011.011.0
11.011.815.216.016.016.014.515.716.517.317.917.716.816.816.816.816.016.0
15.815.812.512.512.512.510.810.811.511.511.511.511.512.215.216.016.016.0
12.514.314.514.713.612.312.212.212.212.212.012.011.211.210.510.510.510.5
10.210.213.513.513.513.513.513.714.214.014.014.012.014.014.014.012.511.0
11.011.011.011.011.011.010.010.010.010.010.010.010.010.014.014.014.014.0
14.014.013.012.012.012.012.014.014.014.012.511.011.011.011.011.011.011.0
10.010.010.010.010.010.010.010.014.014.014.014.014.014.013.012.012.012.0
12.014.014.014.012.511.011.011.011.011.011.011.010.010.010.010.010.010.0
10.010.014.014.014.014.014.014.013.012.012.012.012.014.014.014.012.511.0
11.011.011.011.011.011.010.010.010.010.010.010.010.010.014.014.014.014.0
14.014.013.012.012.012.012.014.014.014.012.511.011.011.011.011.011.011.0
10.010.010.010.010.010.010.010.014.014.014.014.014.014.013.012.012.012.0
12.014.014.014.012.511.011.011.011.011.011.011.010.010.010.010.010.010.0
10.010.014.014.014.014.014.014.013.012.012.012.0
  150
(3(I5,F6.1,F7.1,F6.1))
  183 732.1 4253.9 106.7  182 732.1 4253.9 106.7  180 732.1 4253.9  76.2
  181 732.1 4253.9  76.2  176 733.0 4263.3  34.9  171 738.0 4263.3  20.7
  127 762.6 4266.6  21.3  125 762.8 4266.6  20.7  101 762.6 4266.7  25.9
  108 761.5 4267.6  68.5  178 738.0 4268.3  44.8  174 738.0 4268.3  21.0
  175 738.0 4268.3  25.3  179 738.0 4268.3  61.0  170 738.0 4268.3  20.7
  172 738.0 4268.3  21.0  173 738.0 4268.3  21.0  177 738.0 4268.3  30.5
  148 736.9 4269.4  13.7  131 751.9 4272.8  16.8  165 737.2 4273.2  82.9
  146 742.5 4275.1  68.6  145 742.5 4275.1  68.6  144 742.5 4275.1  68.6
  155 743.0 4275.0  76.2  113 746.4 4275.7  42.7  114 746.4 4275.7  45.7
  110 746.4 4275.7  42.7  111 746.4 4275.7  42.7  112 746.4 4275.7  42.7
  129 746.3 4275.2  15.7  153 743.0 4276.5  45.7  152 743.0 4276.5  45.7
  151 743.0 4276.5  45.7  149 743.0 4276.5  45.7  150 743.0 4276.5  45.7
  118 745.1 4277.3 100.0  119 745.1 4277.3 100.3  120 745.1 4277.3 100.0
  122 745.1 4277.3 100.3  123 745.1 4277.3 100.3  121 745.1 4277.3 100.3
  130 748.5 4277.3  15.2  124 748.3 4277.3   7.6  166 737.2 4278.2  82.9
  109 747.8 4280.9  76.2  160 734.2 4281.0  54.3  156 745.0 4281.0  54.9
  157 745.0 4281.0  55.8  158 745.0 4281.0  56.7  159 745.0 4281.0  56.7
  133 753.0 4281.3  53.3  134 753.0 4281.3  18.6  147 736.8 4283.1  68.6
   15 745.8 4283.5  72.2   14 745.8 4283.5  72.2    8 745.8 4283.5  72.2
   12 745.8 4283.5  72.2   11 745.8 4283.5  72.2   13 745.8 4283.5  72.2
   10 745.8 4283.5  72.2    9 745.8 4283.5  72.2  154 742.5 4284.2  76.2
   84 746.5 4285.7  22.9   83 746.5 4285.7  25.9   94 746.5 4285.7  22.9
  167 725.0 4286.0  15.2   81 745.2 4286.8  21.1   40 747.2 4286.8  14.6
  100 748.9 4286.0   8.3   39 747.0 4287.7  60.0   91 747.0 4287.5  20.0
```

```
MEMBER NAME   FPAGE02
   92 747.0 4287.0   20.0    38 747.0 4287.5   61.0    41 747.8 4287.8   14.6
  168 736.5 4290.5   45.7   169 741.8 4290.5   71.6     7 755.5 4290.1    9.1
   97 753.7 4302.9   45.7    36 754.5 4302.5   54.0    35 754.5 4302.5   24.1
   37 754.5 4302.5   32.7    33 754.5 4302.5   54.9    28 754.5 4302.5   33.5
   34 754.5 4302.5   16.8    29 754.5 4302.5   28.3    30 754.5 4302.5   28.3
   32 754.5 4302.5   30.8    31 754.5 4302.5   27.1    46 754.0 4303.0   95.1
   60 754.0 4303.0  106.7    69 754.0 4303.0   63.1    70 754.0 4303.0   63.1
   73 754.0 4303.0   40.2    74 754.0 4303.0   40.2    59 754.0 4303.0   61.0
   75 754.0 4303.0   42.0    76 754.0 4303.0   42.0    44 754.0 4303.0   56.4
   71 754.0 4303.0   63.1    72 754.0 4303.0   63.1    77 754.0 4303.0   29.8
   50 754.0 4303.0   45.7    43 754.0 4303.0   45.7    49 754.0 4303.0   30.5
   56 754.0 4303.0   45.7    47 754.0 4303.0   45.7    66 754.0 4303.0   45.7
   51 754.0 4303.0   45.7    52 754.0 4303.0   45.7    57 754.0 4303.0   45.7
   64 754.0 4303.0   61.0    65 754.0 4303.0   61.0    68 754.0 4303.0   21.3
   48 754.0 4303.0   45.7    54 754.0 4303.0   45.7    45 754.0 4303.0   45.7
   55 754.0 4303.0   45.7    58 754.0 4303.0   45.7    53 754.0 4303.0   45.7
   61 754.0 4303.0   45.7    62 754.0 4303.0   45.7    63 754.0 4303.0   45.7
   67 754.0 4303.0   45.7     6 748.7 4305.4  106.7     4 748.7 4305.4   76.2
    5 748.7 4305.4   76.2   142 740.4 4306.7  182.9   143 740.4 4306.7  183.0
   93 748.1 4306.7   13.7    26 752.4 4306.6   51.5    16 752.4 4306.6   48.5
   20 752.4 4306.6   48.8    25 752.4 4306.6   23.8    18 752.4 4306.6   48.5
   19 752.4 4306.6   48.5    17 752.4 4306.6   48.5    27 752.4 4306.6   18.4
   24 752.4 4306.6   44.2    21 752.4 4306.6   33.5    23 752.4 4306.6   33.5
   22 752.4 4306.6   33.5    42 746.0 4307.8   51.8    87 747.6 4307.8   58.5
   80 747.6 4307.8   34.1    86 747.6 4307.8   58.3    79 747.6 4307.8   32.0
   85 747.6 4307.8   76.2    78 747.6 4307.8   25.9    88 744.1 4309.7   16.8
(10F7.5,10X)
0.016470.010460.015840.023240.023240.003800.003800.003800.003800.00380
0.013100.013100.013100.013100.013100.004230.004230.004230.004230.00423
0.0    0.0    0.0    0.0    0.0    0.001270.001270.001270.001270.00127
0.017740.010140.012570.022180.036120.015840.015840.015840.015840.01584
0.006550.006550.006550.006550.006550.002120.002120.002120.002120.00212
0.015210.015210.015210.015210.015210.015850.015850.022180.028520.02852
0.019430.015310.017740.029780.054500.027880.027880.027880.027880.02788
0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
0.030420.030420.030420.030420.030420.030420.030420.043090.055770.05577
0.027040.014360.030420.055770.070980.027880.027880.027880.027880.02788
0.208600.208600.0    0.443600.898000.0    0.0    0.0    0.0    0.0
0.030420.030420.030420.030420.030420.030420.030420.043090.055770.05577
0.047320.040980.036330.041830.065910.143700.143700.071830.0    0.0
0.025350.025350.012680.0    0.0    0.029570.029570.029570.029570.02957
0.030420.030420.030420.030420.030420.030420.030420.015850.001270.00127
0.011830.095800.090200.069710.109020.194300.194300.118100.041830.04183
0.006340.006340.032960.059570.059570.029570.029570.029570.029570.02957
0.030420.030420.030420.030420.030420.030420.030420.015850.001270.00127
0.0    0.107740.141100.141330.149600.327400.158400.172350.100480.10048
0.007040.007040.026900.186260.186260.026400.026400.026400.026400.02640
0.016900.016900.038870.060840.060840.015870.015870.013200.681890.38358
0.0    0.078900.136250.158450.131840.322150.454150.179550.215200.07260
0.014080.014080.014080.363100.363100.023240.023240.023240.023240.02324
0.003380.003380.047950.092530.092530.001320.001320.237651.721501.89345
0.0    0.011410.022810.022810.0    0.0    0.216550.338001.748300.17430
0.044600.044600.044600.439700.439700.023240.023240.023240.023240.02324
0.003380.003380.049850.096330.096330.124770.145901.270221.172400.85011
0.0    0.014530.014530.301000.0    0.015850.475302.619501.653002.87850
0.698300.027580.027580.438350.438350.753700.753700.642200.530700.53070
0.375000.375000.149000.149000.149000.258671.106450.722290.186750.31100
0.0    0.047530.068650.0    0.015850.031690.422502.503003.148002.91500
```

```
MEMBER NAME   EPAGEO2
3.517000.316900.010560.430400.430400.558600.558600.454600.350700.35070
0.264101.880000.149000.149000.149000.438330.602280.077530.162450.21630
0.290450.565050.543950.718250.707840.031691.553003.274002.704003.86600
4.236001.363000.316900.084500.084500.363400.363400.267000.170700.17070
1.215000.834400.010560.010560.010561.135450.285180.077130.138200.12170
0.734100.713001.019250.406650.438353.993006.211003.031002.345003.91900
4.468004.922001.912000.084500.084500.363400.363400.267000.170700.17070
1.162000.443600.010560.158400.010560.290480.010560.800850.138200.12170
0.316850.543950.654850.839703.787007.394006.633001.679002.134006.92900
4.151003.855003.961001.098000.316900.031690.242900.010560.010561.90100
0.612600.464701.605000.010561.563001.684500.232400.399050.401350.25690
0.0      0.184850.116200.0      0.316850.940103.972001.415001.595001.21500
5.672005.661003.412003.105000.316900.031690.771102.334001.943001.58400
0.010560.010561.162001.796002.102000.290480.179580.399050.401350.25690
0.248200.0      0.258800.438301.304550.718202.113001.658002.134006.21100
3.158004.690002.715005.408000.718200.116200.369703.264004.352001.88000
2.334001.859000.411900.010560.010560.010560.539820.114570.208950.15630
0.0      0.507000.295750.306300.961103.116002.081000.454203.465001.93300
1.267002.429000.792200.390801.468000.285200.179601.458003.031003.19000
0.559802.081000.422500.010560.781600.002640.006600.003010.016470.05577
0.105630.095050.147850.470050.982350.031690.031690.031691.415001.67900
0.401402.683000.940103.369002.461000.359100.232400.147902.567002.69300
1.394002.493001.722001.162000.010560.002641.029950.006970.016470.05577
0.063350.073950.195400.258750.269350.549200.031690.686604.130005.58800
2.746004.943007.341004.119000.918900.031690.031690.147900.147900.14790
1.753001.299001.711001.838000.010560.145300.036330.148450.296900.01056
0.079200.095050.084500.385500.353803.803008.270004.552003.264005.31300
6.591005.059005.091005.355002.514000.031690.380200.147900.147900.14790
0.010560.876700.676000.010560.010560.145300.036330.148450.296900.01056
0.063350.195400.411900.433050.617903.000003.602005.619004.647005.65100
4.014005.915004.975003.729001.204000.052810.052810.147900.147900.14790
0.038730.038730.038730.580900.010560.142000.058940.164080.296900.01056
0.0      0.253500.242950.390800.443600.464706.179005.693006.010004.33100
4.394004.236005.186001.796000.718200.052810.749900.612600.010560.01056
0.038730.038730.038730.010560.010560.138600.081560.696761.352150.01056
0.0      0.153150.195400.279900.190150.190100.211204.035002.060003.33800
3.919004.605002.841000.644300.316900.137301.341001.194000.010560.01056
0.038730.038730.038730.010560.549200.138600.081560.638760.591500.15313
0.058100.390850.316900.153150.184850.401400.580900.929500.982303.29600
1.436000.031690.570400.084500.042250.052811.817002.662000.876700.01056
0.010560.010560.010560.010560.010560.002110.002110.265101.082850.74460
0.184850.385550.290450.364400.501750.623200.633700.496402.852001.84800
0.918900.475300.348600.044890.044890.295700.052811.141000.295700.01056
0.010560.010560.010560.866100.010560.002110.002110.075000.808150.27988
0.190100.438300.258800.279900.158450.253500.380200.422502.820000.79220
0.771100.654900.316900.044890.044890.010561.172001.162002.683002.56700
1.500000.010560.010560.010560.010560.002110.002110.002810.003520.37498
0.063350.522800.443650.311550.253500.316900.327400.232400.380202.56700
2.820000.179600.316900.063370.063370.023770.023771.077002.070001.35200
2.947000.813300.010560.010560.010560.002110.002110.002810.003520.05808
0.184850.327450.470050.353800.332700.295700.137300.147900.792200.29570
0.137300.031690.316900.063370.063370.023770.023770.031690.813300.99290
1.183001.162000.010560.327400.010560.002110.002110.002810.003520.58628
0.242950.522850.100350.264050.248200.137300.0      0.411900.158400.10560
0.126700.253500.316900.031690.031690.005280.005280.095060.116200.01056
2.218000.739402.387000.010561.806000.048160.048160.057380.067600.06760
0.0      0.586100.950600.390800.338000.665400.338000.274600.475300.43310
0.169000.137300.306300.316900.200700.005280.005280.010560.084500.01056
```

```
MEMBER NAME   EPAGE02
0.728801.426000.010560.866100.105600.048160.048160.057880.067600.06760
0.182170.509621.423200.330070.250850.147850.216500.433050.480600.25875
0.311600.237650.348550.422500.174300.016430.016430.016430.010560.01056
0.010560.095080.010560.337980.010560.048160.048160.057880.067600.06760
0.113200.144400.088820.298400.340650.073950.0      0.285200.327450.0
0.092950.092950.075200.057450.057450.037150.037150.037150.034220.03422
0.688470.012250.012250.329120.012250.036750.036750.048800.057460.04732
0.074780.193600.116600.406720.488500.211250.0      0.285200.0      0.22710
0.151700.151700.120830.089980.099980.057880.057880.057880.057880.05788
0.013940.013940.013940.013940.013940.025350.025350.035070.043940.11534
0.0      0.209150.439150.437000.258770.343250.0      0.0      0.116200.11620
0.049010.049010.032110.015210.015210.057880.057830.057880.057880.05788
0.013940.013940.013940.013940.013940.025350.025350.021120.019930.50504
0.037180.037180.023020.008870.008870.016480.016480.016480.016480.01648
0.013100.013100.013100.013100.013100.016900.016900.010140.003380.00338
0.044890.044890.396080.010560.058080.339210.002640.023290.196670.68732
0.037180.037180.023020.008870.008870.016480.016480.016480.016480.01648
0.013100.013100.013100.013100.013100.004230.004230.010770.017320.01732
0.084500.063380.374980.205980.480580.006600.002640.008080.011430.00517
0.033800.033800.027460.021120.021120.021760.021760.021760.021760.02176
0.013590.013590.013590.011830.011830.005280.005280.047740.016260.01626
0.010560.602050.697281.568500.306300.213880.003520.010840.018160.01816
0.030420.030420.031900.033380.033380.027040.027040.027040.027040.02704
0.014080.014080.014080.047530.010560.438350.089780.100320.010560.04753
1.510501.769001.341350.010560.749780.042240.007040.019990.032950.03295
0.030420.030420.031900.033380.033380.027040.027040.027040.027040.02704
0.044890.044890.068651.616501.014002.403000.522781.964501.843500.78155
0.010560.010560.047530.010561.040280.168990.124100.217160.032950.03295
0.078160.078160.116830.116620.0      0.065490.065490.065490.065490.06549
0.256100.256100.396080.987800.961150.227080.126730.712800.855470.32743
0.064220.064220.214600.365000.365001.122240.021120.012670.014790.01479
 886.46484
(9F8.2)
 1174.17 1127.57  515.01  196.74    7.11   37.66    1.27    1.24    1.04
   27.07   49.80   45.16   43.29   31.21   28.91   14.87   14.87   13.87
  154.19    0.72   39.35   46.54    9.32    6.47    3.14   91.36   87.94
   70.88   67.72   22.58    0.20  100.63   40.79   37.54   20.31   12.28
   36.59   36.59   36.59   12.34   11.33    4.34    0.55    0.12   49.19
    1.50   13.35  134.34   79.40   70.19   68.46   52.07    7.48   41.60
  332.40  263.59   68.92   67.03   66.39   57.50   54.60   52.70    3.14
  195.33  168.54    0.03    0.49    3.28   15.53    0.5815908.64    5.55
    5.55    4.55    2.24    1.35  135.78    2.65   74.10   63.52   15.07
   11.51    9.38    6.90    6.36    6.30    6.30    6.18    5.61   96.63
   80.78   46.20   46.20   46.20   46.20   43.01   42.72   42.72   42.63
   31.38   31.38   30.23   29.37   26.32   18.96   14.79   12.80   12.66
   11.77   11.77   11.54    9.98    9.98    9.26    9.15    8.72    8.66
    7.97    5.98    5.67    5.38    5.39    5.38    4.57 1720.91  693.51
  278.66 1681.87 1640.59    0.06   91.94   51.72   48.39   42.00   36.99
   36.99   25.86   13.20   12.66    5.24    4.63    4.55    5.72  242.19
  149.47   82.04   56.93   51.58    7.05    3.34
   30286.6
(12F6.1)
1062.51013.4 730.7 730.7  84.0 233.3   0.0  32.6   0.0 259.8  94.4 233.7
 228.4 146.1 189.6 193.9 193.9 149.3   0.0   0.0 218.8 171.6 319.5 185.0
 550.3 209.7 209.0 122.0 118.9 121.7  37.2 198.9 126.7 121.5 114.7 113.9
 943.1 942.0 881.11185.31149.7 542.1   0.0   0.0 370.4   0.0 153.1 986.6
 503.4 503.4 503.4  59.8  55.3 469.7 558.2 558.2 397.7 386.1 386.1 385.9
 386.1 397.7 550.3 270.7 260.6   0.0 190.6  98.6  78.6  75.1 300.0  40.0
```

MEMBER NAME EPAGE02

40.0	88.0	70.0	101.5	720.8	768.8	210.4	261.0	201.5	190.6	216.8	195.6
176.8	261.8	234.7	238.2	183.1	704.6	858.9	334.7	334.7	324.9	324.9	632.1
325.0	325.0	367.6	270.7	270.7	163.0	605.7	304.0	495.3	391.4	325.9	371.7
301.0	301.0	335.4	473.7	473.7	293.7	239.5	462.9	150.0	314.9	237.8	235.5
282.9	282.9	278.2	118.2	779.2	585.3	433.4	1210.0	1210.0	0.0	793.1	277.0
383.6	145.3	240.8	240.8	195.9	116.4	50.0	311.2	311.2	311.2	385.0	533.3
231.3	208.5	100.0	277.4	73.8	57.8						

4. Output Sample

```
&INLIST
     IM=30, JM=40, KM=14,IN=9,JN=13,KN=1,KNN=10,NS=25,LM=150,
     IMC=15,JMC=19,
     DX=5*2000.,20*1000.,5*2000.,DY=10*2000.,20*1000.,10*2000.,
     DZ=5*20.,9*25.,TM=0.0, DT=120.,AKA=24*1.0E-4,AKH=0.,13*10.,
     HS=10., HP=140., HG=1000., ZMAX=1000.,HMIN=300.,HMAX=600.,
     ZRPQ=0.6,ZRISE=1.0,PMAX=0.5,PMIN=0.15,DCMIN=2.0,OLMIN=30.,
     IHR=1,IDAY=1,IMO= 2,IYR=75, LTSTOP=4,IHRTP=1,IDAYTP=1,
     JUNIT=6,KUNITG=13,KUNITS=14,KUNITW=15,KUNITC=16,KUNITP=17,
     LCRUN=1,LHJUS=0,LCHEM=1,LWW=0,LTOP=1,LWTOP=1,LSOUS=1,1,
     LWIND=2,KWIND=4,LPQ=1,LTSOUS= 3600,LTWIND= 3600,ZOMEAN=0.0,
     LWRITE=1,1,1,1,2,2,1,2,0,0,
&END

CARDS COPIED TO DISK.
```

***** LOCATION OF 25 RAMS STATIONS *****

IS	XSUTM	YSUTM	XS	YS	JXS	JYS
1	744.18	4279.86	14.18	17.86	15	18
2	742.52	4286.04	12.52	24.04	13	25
3	747.59	4282.46	17.59	20.46	18	21
4	747.31	4277.30	17.31	15.30	18	16
5	743.71	4276.45	13.71	14.45	14	15
6	738.66	4277.57	8.66	15.57	9	16
7	740.18	4282.61	10.18	20.61	11	21
8	748.41	4291.10	18.41	29.10	19	30
9	755.80	4279.89	25.40	17.89	26	18
10	747.21	4272.82	17.21	10.82	18	11
11	738.81	4272.48	8.81	10.48	9	11
12	733.94	4280.91	4.47	18.91	5	19
13	737.74	4289.82	7.74	27.82	8	28
14	744.32	4297.46	14.32	32.73	15	33
15	757.11	4297.80	26.06	32.90	27	33
16	762.78	4290.08	28.89	28.08	29	29
17	760.56	4272.82	27.78	10.82	28	11
18	743.06	4263.26	13.06	5.63	14	6
19	729.76	4270.55	2.38	9.27	3	10
20	723.08	4285.91	-0.96	23.91	1	24
21	732.41	4302.37	3.71	35.19	4	36
22	741.63	4329.22	11.63	48.61	12	40
23	777.32	4286.38	36.16	24.38	30	25
24	749.27	4236.54	19.27	-7.73	20	1
25	697.45	4282.24	-13.78	20.24	1	21

*** ZS FIELD UNIT IN M. , RATIO= 1.00

..... PAGE = 1

	XUTM=	725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4308	39	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4306	38	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4304	37	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4302	36	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4300	35	12	14	14	14	12	11	11	11	11	11	11	11	10	10	10	10	10	10	10	10	14	14	14	14	14	14	13	12	12	12
4298	34	12	14	14	14	13	12	12	12	12	12	12	12	11	11	10	10	10	10	10	10	13	13	13	13	13	13	14	14	14	14
4296	33	14	15	16	17	17	17	16	16	16	16	16	16	15	15	12	12	12	12	10	10	11	11	11	11	11	12	15	16	16	16
4294	32	15	16	17	18	19	19	18	18	18	13	17	17	17	17	13	13	13	13	11	11	11	11	11	11	11	11	15	16	16	16
4292	31	16	16	16	18	18	19	18	18	18	18	18	17	17	16	13	12	12	12	11	11	11	11	11	11	11	12	15	16	16	16
4291	30	20	16	15	18	18	20	20	20	20	21	21	20	20	13	13	11	11	11	11	14	14	13	12	12	12	12	15	16	16	16
4290	29	20	16	15	18	18	20	20	20	20	21	21	20	20	13	13	11	11	11	11	14	14	13	12	12	12	12	14	15	15	15
4289	28	20	19	18	19	19	20	20	20	20	20	21	19	14	10	10	12	15	15	15	18	18	15	12	12	12	12	14	15	15	15
4288	27	20	20	20	20	20	20	20	20	20	20	20	18	13	10	10	14	20	19	18	18	18	14	11	11	11	11	14	15	15	15
4287	26	20	20	20	20	19	19	20	20	20	20	20	18	13	10	12	17	20	19	17	15	15	13	11	11	11	11	14	15	15	15
4286	25	20	20	20	20	19	19	20	21	21	20	20	21	16	10	15	21	21	18	15	15	15	13	11	11	11	11	14	15	15	15
4285	24	20	20	21	20	20	20	21	21	21	20	20	21	16	11	16	21	20	17	13	11	11	11	11	11	11	11	14	15	15	15
4284	23	20	20	20	20	20	20	20	20	20	20	20	20	21	19	18	21	20	18	13	11	11	11	12	12	12	12	14	15	15	15
4283	22	21	20	20	20	20	20	20	20	20	20	20	20	21	20	18	19	21	19	15	14	14	13	12	12	12	12	14	15	15	15
4282	21	21	20	20	20	20	20	20	20	20	20	20	20	20	20	18	18	21	21	17	14	14	13	12	12	12	11	11	11	11	11
4281	20	20	20	20	20	19	18	21	23	21	20	20	20	20	20	18	18	22	22	19	18	18	15	12	12	12	11	11	11	11	11
4280	19	20	20	18	20	19	19	20	21	21	20	21	20	20	21	18	18	22	22	19	18	18	16	15	15	15	14	11	11	11	11
4279	18	20	20	19	20	20	20	21	21	21	22	22	22	22	22	19	19	21	20	19	19	19	17	15	15	15	14	11	11	11	11
4278	17	20	20	20	20	20	20	22	22	22	21	21	21	21	22	19	18	20	20	19	19	19	17	15	15	15	14	11	11	11	11
4277	16	20	20	20	21	20	20	21	20	20	20	20	20	20	21	19	18	20	20	19	19	19	17	15	15	15	14	11	11	11	11
4276	15	20	20	20	20	20	20	20	21	20	20	20	20	20	19	18	20	20	20	19	19	19	17	16	16	16	15	11	11	11	11
4275	14	20	20	20	20	20	20	20	21	20	20	20	20	20	17	17	20	19	18	18	18	18	17	16	16	16	15	11	11	11	11
4274	13	20	20	20	20	20	20	20	20	21	20	20	20	17	17	20	21	16	13	16	18	18	17	16	16	16	15	14	14	14	14
4273	12	20	20	20	20	20	20	20	20	20	20	20	16	12	14	19	19	14	11	14	16	16	16	16	16	16	15	14	14	14	14
4272	11	20	20	20	20	20	20	20	20	20	20	16	11	10	12	17	19	16	12	13	16	16	15	15	15	15	14	14	14	14	14
4270	10	20	20	20	20	20	20	20	20	19	18	14	12	13	15	14	14	14	13	14	14	14	14	15	15	15	14	14	14	14	14
4268	9	20	20	20	20	20	19	19	18	16	13	12	13	14	18	18	14	14	11	11	12	12	12	13	13	13	13	13	14	14	14
4266	8	20	20	20	20	20	17	17	15	15	11	11	14	14	19	19	15	15	10	10	10	10	11	12	12	12	12	13	14	14	14
4264	7	20	20	20	20	20	19	15	15	15	14	11	11	11	13	13	12	12	10	10	10	10	11	11	11	11	11	13	14	14	14
4262	6	20	20	21	20	20	20	15	15	15	15	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	13	14	14	14
4260	5	19	20	20	20	19	18	14	14	14	14	11	11	10	10	10	10	10	10	10	10	10	10	10	10	10	11	13	14	14	14
4258	4	19	18	18	18	15	13	12	12	12	12	11	11	10	10	10	10	10	10	10	10	10	10	10	10	10	10	11	12	12	12
4256	3	19	18	18	18	15	12	12	12	12	12	12	12	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
4254	2	19	18	18	18	15	12	12	12	12	12	12	12	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
4252	1	19	18	18	18	15	12	12	12	12	12	12	12	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10

*** ZO FIELD UNIT IN M. , RATIO= 100.00
..... PAGE = 1

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	23	29	31	31	25	25	25	25	25	25	31	31	23	35	33	18	17	29	31	20	25	25	29	33	33	56	24	24	24	24
4308	39	20	26	27	27	25	25	25	25	25	25	25	25	16	47	35	62	25	54	51	30	23	23	24	23	55	27	26	31	25	25
4306	38	20	26	27	27	25	25	25	25	25	25	24	24	14	16	14	24	17	17	14	16	68	76	63	23	45	24	23	24	25	25
4304	37	22	27	26	26	25	25	25	25	25	25	24	24	14	14	14	14	14	16	14	14	23	41	43	70	31	29	23	24	24	24
4302	36	22	27	26	26	25	25	25	25	25	25	24	24	14	14	14	14	14	14	14	14	25	25	33	29	37	23	23	24	24	24
4300	35	22	27	26	26	25	25	25	25	25	25	24	24	14	14	14	14	14	14	14	14	24	24	35	23	25	32	23	24	30	45
4298	34	23	36	44	45	38	41	26	26	31	31	27	27	18	18	16	17	17	17	16	16	22	22	22	22	22	23	27	30	30	48
4296	33	36	43	43	62	70	55	37	52	37	50	41	41	37	36	22	20	20	20	16	16	17	17	17	17	17	19	31	36	36	38
4294	32	39	43	43	60	68	50	41	56	58	41	41	41	39	38	22	22	22	22	16	16	29	14	14	22	14	18	31	36	37	36
4292	31	49	66	117	63	61	55	54	66	68	55	52	48	52	50	25	19	19	19	14	17	17	18	16	24	16	19	32	37	37	37
4291	30	52	81	99	73	62	85	64	60	72	66	50	43	50	25	23	14	14	14	17	22	43	56	17	37	19	20	32	37	37	37
4290	29	70	77	49	64	57	52	44	68	52	48	49	48	50	18	18	14	14	17	18	22	86	38	73	17	58	22	33	37	37	37
4289	28	70	76	86	79	75	73	61	57	93	56	45	35	31	12	12	17	25	25	50	68	76	60	17	25	17	19	31	35	35	62
4288	27	61	101	89	79	75	76	73	62	64	171	172	37	24	11	11	23	33	80	114	83	147	47	14	14	14	18	31	35	35	37
4287	26	72	97	79	79	68	74	81	75	210	86	82	62	25	11	16	27	87	83	131	103	69	18	14	14	14	18	31	35	35	52
4286	25	72	87	79	82	92	100	100	85	227	153	95	64	37	12	23	49	37	76	35	24	24	18	14	32	14	18	31	38	74	49
4285	24	63	91	80	64	63	81	94	114	109	232	126	44	50	16	25	37	117	128	43	14	14	14	14	14	14	18	31	48	87	70
4284	23	60	74	70	68	57	61	62	297	164	230	268	314	202	57	39	39	93	75	19	14	16	17	18	17	30	24	36	66	62	42
4283	22	61	81	68	69	68	72	417	385	397	293	290	264	303	117	54	31	68	56	24	20	20	19	18	17	17	24	36	68	100	36
4282	21	66	72	75	72	79	203	243	369	318	376	272	364	305	231	79	29	37	41	31	24	20	19	18	31	17	23	29	36	41	30
4281	20	55	57	51	68	62	227	502	328	239	335	419	338	339	331	128	29	54	43	37	33	29	50	32	17	17	23	29	35	41	30
4280	19	54	52	52	58	56	72	43	80	267	348	206	342	489	277	68	30	37	43	37	33	95	70	76	81	24	29	30	35	41	30
4279	18	56	51	50	69	101	44	44	44	123	143	70	211	100	225	132	43	44	39	140	144	89	118	76	60	24	24	68	29	30	31
4278	17	50	75	61	60	99	222	175	75	259	161	122	189	91	62	91	37	39	94	160	165	55	103	36	24	48	24	27	29	30	31
4277	16	70	47	56	68	117	87	181	151	173	417	239	331	206	331	56	31	48	172	218	110	130	95	36	24	24	25	49	33	37	35
4276	15	52	62	49	41	57	95	286	139	143	115	406	409	251	181	38	31	66	134	114	98	31	29	66	87	98	37	36	44	44	38
4275	14	81	93	88	89	239	456	442	155	177	448	293	278	256	64	36	33	39	30	30	106	54	44	81	25	80	86	37	44	44	38
4274	13	111	109	119	66	61	244	378	221	185	277	310	298	99	27	36	50	37	27	31	36	75	43	25	31	25	37	36	77	44	43
4273	12	79	98	92	83	75	38	118	213	189	260	259	70	20	22	32	47	31	19	26	32	69	55	25	25	25	69	49	42	44	43
4272	11	56	58	60	44	42	38	57	161	190	175	147	18	12	26	41	52	44	30	31	38	36	103	37	37	37	51	67	42	45	49
4270	10	56	55	50	69	42	37	56	152	102	138	36	16	19	36	35	43	44	39	39	41	36	42	37	37	37	43	92	74	47	52
4268	9	51	54	50	44	42	35	43	42	82	22	17	22	25	56	58	31	31	22	20	23	23	26	31	33	33	37	43	102	97	81
4266	8	48	56	60	57	55	50	56	27	29	16	14	25	25	66	66	37	37	24	24	24	24	25	31	31	31	30	35	47	118	127
4264	7	45	55	57	54	51	57	31	29	25	24	20	24	24	35	42	32	32	30	33	33	33	33	31	31	31	31	33	35	64	51
4262	6	45	52	54	49	48	52	30	26	25	25	23	23	24	25	32	31	31	31	37	37	37	37	31	31	31	31	35	35	33	33
4260	5	48	47	48	44	43	45	26	25	23	23	23	23	24	24	30	31	31	31	36	36	37	37	31	31	31	31	35	35	33	33
4258	4	49	35	36	37	31	24	19	19	19	19	25	25	26	43	63	29	29	29	30	30	31	31	30	30	30	30	31	31	32	32
4256	3	49	32	32	32	26	19	19	19	19	19	18	18	27	27	27	27	27	27	27	27	30	30	30	30	30	30	29	29	29	29
4254	2	49	31	32	32	26	18	18	18	18	18	18	18	29	29	29	27	27	27	27	27	29	29	29	29	29	29	27	27	27	27
4252	1	48	31	32	32	25	18	18	18	18	18	18	18	29	29	29	29	29	29	29	29	27	27	27	27	27	27	27	26	26	26

*** QB FIELD UNIT IN G/S , RATIO= 1.00
..... PAGE = 1

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	0	0	0	0	0	0	0	0	0	0	1	1	1	3	3	0	0	2	3	1	0	0	0	1	1	4	0	0	0	0
4308	39	0	0	0	0	0	0	0	0	0	0	0	0	0	6	4	9	2	7	7	3	0	0	0	0	4	0	0	0	0	0
4306	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	6	7	5	0	2	0	0	0	0	0
4304	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	6	1	0	0	0	0	0
4302	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
4300	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	2
4298	34	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
4296	33	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4294	32	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	0	0	0	0	0
4292	31	0	2	5	1	1	0	0	1	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
4291	30	0	2	3	1	1	2	1	1	1	1	0	0	1	1	0	0	0	0	0	0	2	5	0	3	0	0	0	0	0	0
4290	29	0	2	0	1	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	8	2	9	0	7	0	0	0	0	0
4289	28	0	1	1	1	1	1	0	0	3	1	0	0	1	0	0	0	0	0	3	3	4	4	0	1	0	0	0	0	0	2
4288	27	0	2	1	1	1	1	1	0	1	10	11	0	1	0	0	0	0	4	8	5	11	3	0	0	0	0	0	0	0	0
4287	26	0	1	1	1	0	1	1	1	11	3	3	2	1	0	0	0	4	4	10	10	6	0	0	0	0	0	0	0	0	1
4286	25	0	1	1	1	2	2	2	1	11	7	3	1	1	0	0	1	0	4	1	0	0	0	0	3	0	0	0	0	3	1
4285	24	0	1	1	0	0	1	2	3	3	13	5	0	2	0	0	0	7	10	3	0	0	0	0	0	0	0	0	1	4	2
4284	23	0	0	0	1	0	0	0	16	8	13	15	18	11	2	1	0	5	4	0	0	0	0	0	0	2	0	0	2	2	0
4283	22	0	1	0	1	1	1	24	22	24	17	17	16	20	7	2	0	2	2	0	0	0	0	0	0	0	0	0	2	5	0
4282	21	0	0	1	1	2	12	14	22	18	22	16	23	19	14	4	0	0	0	0	0	0	0	0	2	0	0	0	0	1	0
4281	20	0	0	0	1	1	15	33	18	13	21	26	20	20	21	10	0	1	0	0	0	0	3	2	0	0	0	0	0	1	0
4280	19	0	0	0	1	1	2	0	2	16	22	10	19	29	16	3	0	0	0	0	0	7	5	6	7	0	0	0	0	1	0
4279	18	0	0	0	1	3	0	0	0	5	6	1	10	3	13	9	1	0	0	10	10	5	9	6	4	0	0	4	0	0	0
4278	17	0	2	1	1	3	12	8	1	13	7	5	9	3	1	5	1	0	5	12	12	2	8	1	0	3	0	0	0	0	0
4277	16	0	0	1	1	5	2	8	6	8	24	12	18	10	21	2	0	1	13	17	7	9	7	1	0	0	0	2	0	0	0
4276	15	0	0	0	0	1	3	15	5	6	4	22	22	13	12	1	0	3	9	7	6	0	0	4	7	8	1	0	1	1	1
4275	14	1	2	2	3	15	29	26	6	8	27	16	15	15	4	1	0	0	0	0	7	2	1	6	0	6	6	0	1	1	1
4274	13	2	2	4	1	1	15	24	12	9	15	17	19	7	0	0	1	1	1	0	0	4	1	0	0	0	1	0	3	0	0
4273	12	1	2	2	2	2	0	6	13	10	15	16	5	1	0	0	1	1	1	0	0	4	3	0	0	0	4	1	0	0	0
4272	11	0	0	0	0	0	0	1	10	12	11	14	1	0	1	1	2	2	1	1	1	1	7	0	0	0	1	2	0	0	0
4270	10	0	0	0	1	0	0	1	10	6	11	2	0	0	1	1	3	3	2	2	2	1	1	0	0	0	1	4	2	0	1
4268	9	0	0	0	0	0	0	0	1	6	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	5	4	3
4266	8	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7
4264	7	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1
4262	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4260	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4258	4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4256	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4254	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4252	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

.... TOTAL EMISSION : READ IN= 886.5, MODEL USED= 3545.9, IN G/SEC

*** POINT SOURCE DATA ***
..... TOTAL NUMBER = 150

N	ID	XPUTM	YPUTM	ZP	QP	ZR	XP	YP
1	183	732.1	4253.9	106.7	1174.17	1062.5	3.55	0.95
2	182	732.1	4253.9	106.7	1127.57	1013.4	3.55	0.95
3	180	732.1	4253.9	76.2	515.01	730.7	3.55	0.95
4	181	732.1	4253.9	76.2	196.74	730.7	3.55	0.95
5	176	733.0	4263.3	34.9	7.11	84.0	4.00	5.65
6	171	738.0	4263.3	20.7	37.66	233.3	8.00	5.65
7	127	762.6	4266.6	21.3	1.27	0.0	28.80	7.30
8	125	762.8	4266.6	20.7	1.24	32.6	28.90	7.30
9	101	762.6	4266.7	25.9	1.04	0.0	28.80	7.35
10	108	761.5	4267.6	68.5	27.07	259.8	28.25	7.80
11	178	738.0	4268.3	44.8	49.80	94.4	8.00	8.15
12	174	738.0	4268.3	21.0	45.16	233.7	8.00	8.15
13	175	738.0	4268.3	25.3	43.29	228.4	8.00	8.15
14	179	738.0	4268.3	61.0	31.21	146.1	8.00	8.15
15	170	738.0	4268.3	20.7	28.91	189.6	8.00	8.15
16	172	738.0	4268.3	21.0	14.87	193.9	8.00	8.15
17	173	738.0	4268.3	21.0	14.87	193.9	8.00	8.15
18	177	738.0	4268.3	30.5	13.87	149.3	8.00	8.15
19	148	736.9	4269.4	13.7	154.19	0.0	6.90	8.70
20	131	751.9	4272.8	16.8	0.72	0.0	21.90	10.80
21	165	737.2	4273.2	82.9	39.35	218.8	7.20	11.20
22	146	742.5	4275.1	68.6	46.54	171.6	12.50	13.10
23	145	742.5	4275.1	68.6	9.32	319.5	12.50	13.10
24	144	742.5	4275.1	68.6	6.47	185.0	12.50	13.10
25	155	743.0	4275.0	76.2	3.14	550.3	13.00	13.00
26	113	746.4	4275.7	42.7	91.36	209.7	16.40	13.70
27	114	746.4	4275.7	45.7	87.94	209.0	16.40	13.70
28	110	746.4	4275.7	42.7	70.88	122.0	16.40	13.70
29	111	746.4	4275.7	42.7	67.72	118.9	16.40	13.70
30	112	746.4	4275.7	42.7	22.58	121.7	16.40	13.70
31	129	746.3	4275.2	15.7	0.20	37.2	16.30	13.20
32	153	743.0	4276.5	45.7	100.63	198.9	13.00	14.50
33	152	743.0	4276.5	45.7	40.79	126.7	13.00	14.50
34	151	743.0	4276.5	45.7	37.54	121.5	13.00	14.50
35	149	743.0	4276.5	45.7	20.31	114.7	13.00	14.50
36	150	743.0	4276.5	45.7	12.28	113.9	13.00	14.50
37	118	745.1	4277.3	100.0	36.59	943.1	15.10	15.30
38	119	745.1	4277.3	100.3	36.59	942.0	15.10	15.30
39	120	745.1	4277.3	100.0	36.59	881.1	15.10	15.30
40	122	745.1	4277.3	100.3	12.34	1185.3	15.10	15.30
41	123	745.1	4277.3	100.3	11.33	1149.7	15.10	15.30
42	121	745.1	4277.3	100.3	4.34	542.1	15.10	15.30
43	130	748.5	4277.3	15.2	0.55	0.0	18.50	15.30
44	124	748.3	4277.3	7.6	0.12	0.0	18.30	15.30
45	166	737.2	4278.2	82.9	49.19	370.4	7.20	16.20
46	109	747.8	4280.9	76.2	1.50	0.0	17.80	18.90
47	160	734.2	4281.0	54.3	13.35	153.1	4.60	19.00
48	156	745.0	4281.0	54.9	134.34	986.6	15.00	19.00
49	157	745.0	4281.0	55.8	79.40	503.4	15.00	19.00
50	158	745.0	4281.0	56.7	70.19	503.4	15.00	19.00
51	159	745.0	4281.0	56.7	68.46	503.4	15.00	19.00
52	133	753.0	4281.3	53.3	52.07	59.8	23.00	19.30
53	134	753.0	4281.3	18.6	7.48	55.3	23.00	19.30
54	147	736.8	4283.1	68.6	41.60	469.7	6.80	21.10
55	15	745.8	4283.5	72.2	332.40	558.2	15.80	21.50
56	14	745.8	4283.5	72.2	263.59	558.2	15.80	21.50
57	8	745.8	4283.5	72.2	68.92	397.7	15.80	21.50
58	12	745.8	4283.5	72.2	67.03	386.1	15.80	21.50
59	11	745.8	4283.5	72.2	66.39	386.1	15.80	21.50
60	13	745.8	4283.5	72.2	57.50	385.9	15.80	21.50
61	10	745.8	4283.5	72.2	54.60	386.1	15.80	21.50
62	9	745.8	4283.5	72.2	52.70	397.7	15.80	21.50
63	154	742.5	4284.2	76.2	3.14	550.3	12.50	22.20
64	84	746.5	4285.7	22.9	195.33	270.7	16.50	23.70
65	83	746.5	4285.7	25.9	168.54	260.6	16.50	23.70
66	94	746.5	4285.7	22.9	0.03	0.0	16.50	23.70
67	167	725.0	4286.0	15.2	0.49	190.6	0.0	24.00
68	81	745.2	4286.8	21.1	3.28	98.6	15.20	24.80
69	40	747.2	4286.8	14.6	15.53	78.6	17.20	24.80
70	100	748.9	4286.0	8.3	0.58	75.1	18.90	24.00
71	39	747.0	4287.7	60.0	0.0	300.0	17.00	25.70
72	91	747.0	4287.5	20.0	5.55	40.0	17.00	25.50
73	92	747.0	4287.0	20.0	5.55	40.0	17.00	25.00
74	38	747.0	4287.5	61.0	4.55	88.0	17.00	25.50
75	41	747.8	4287.8	14.6	2.24	70.0	17.80	25.80

76	168	736.5	4290.5	45.7	1.35	101.5	6.50	28.50
77	169	741.8	4290.5	71.6	135.78	720.8	11.80	28.50
78	7	755.5	4290.1	9.1	2.65	768.8	25.25	28.10
79	97	753.7	4302.9	45.7	74.10	210.4	23.70	35.45
80	36	754.5	4302.5	54.0	63.52	261.0	24.50	35.25
81	35	754.5	4302.5	24.1	15.07	201.5	24.50	35.25
82	37	754.5	4302.5	32.7	11.51	190.6	24.50	35.25
83	33	754.5	4302.5	54.9	9.38	216.8	24.50	35.25
84	28	754.5	4302.5	33.5	6.90	195.6	24.50	35.25
85	34	754.5	4302.5	16.8	6.36	176.8	24.50	35.25
86	29	754.5	4302.5	28.3	6.30	261.8	24.50	35.25
87	30	754.5	4302.5	28.3	6.30	234.7	24.50	35.25
88	32	754.5	4302.5	30.8	6.18	238.2	24.50	35.25
89	31	754.5	4302.5	27.1	5.61	183.1	24.50	35.25
90	46	754.0	4303.0	95.1	96.63	704.6	24.00	35.50
91	60	754.0	4303.0	106.7	80.78	858.9	24.00	35.50
92	69	754.0	4303.0	63.1	46.20	334.7	24.00	35.50
93	70	754.0	4303.0	63.1	46.20	334.7	24.00	35.50
94	73	754.0	4303.0	40.2	46.20	324.9	24.00	35.50
95	74	754.0	4303.0	40.2	46.20	324.9	24.00	35.50
96	59	754.0	4303.0	61.0	43.01	632.1	24.00	35.50
97	75	754.0	4303.0	42.0	42.72	325.0	24.00	35.50
98	76	754.0	4303.0	42.0	42.72	325.0	24.00	35.50
99	44	754.0	4303.0	56.4	42.63	367.6	24.00	35.50
100	71	754.0	4303.0	63.1	31.38	270.7	24.00	35.50
101	72	754.0	4303.0	63.1	31.38	270.7	24.00	35.50
102	77	754.0	4303.0	29.8	30.23	163.0	24.00	35.50
103	50	754.0	4303.0	45.7	29.37	605.7	24.00	35.50
104	43	754.0	4303.0	45.7	26.32	304.0	24.00	35.50
105	49	754.0	4303.0	30.5	18.96	495.3	24.00	35.50
106	56	754.0	4303.0	45.7	14.79	391.4	24.00	35.50
107	47	754.0	4303.0	45.7	12.80	325.9	24.00	35.50
108	66	754.0	4303.0	45.7	12.66	371.7	24.00	35.50
109	51	754.0	4303.0	45.7	11.77	301.0	24.00	35.50
110	52	754.0	4303.0	45.7	11.77	301.0	24.00	35.50
111	57	754.0	4303.0	45.7	11.54	335.4	24.00	35.50
112	64	754.0	4303.0	61.0	9.98	473.7	24.00	35.50
113	65	754.0	4303.0	61.0	9.98	473.7	24.00	35.50
114	68	754.0	4303.0	21.3	9.26	293.7	24.00	35.50
115	48	754.0	4303.0	45.7	9.15	239.5	24.00	35.50
116	54	754.0	4303.0	45.7	8.72	462.9	24.00	35.50
117	45	754.0	4303.0	45.7	8.66	150.0	24.00	35.50
118	55	754.0	4303.0	45.7	7.97	314.9	24.00	35.50
119	58	754.0	4303.0	45.7	5.98	237.8	24.00	35.50
120	53	754.0	4303.0	45.7	5.67	235.5	24.00	35.50
121	61	754.0	4303.0	45.7	5.38	282.9	24.00	35.50
122	62	754.0	4303.0	45.7	5.38	282.9	24.00	35.50
123	63	754.0	4303.0	45.7	5.38	278.2	24.00	35.50
124	67	754.0	4303.0	45.7	4.57	118.2	24.00	35.50
125	6	748.7	4305.4	106.7	1720.91	779.2	18.70	36.70
126	4	748.7	4305.4	76.2	693.51	585.3	18.70	36.70
127	5	748.7	4305.4	76.2	278.66	433.4	18.70	36.70
128	142	740.4	4306.7	182.9	1681.87	1210.0	10.40	37.35
129	143	740.4	4306.7	183.0	1640.59	1210.0	10.40	37.35
130	93	748.1	4306.7	13.7	0.06	0.0	18.10	37.35
131	26	752.4	4306.6	51.5	91.94	793.1	22.40	37.30
132	16	752.4	4306.6	48.5	51.72	277.0	22.40	37.30
133	20	752.4	4306.6	48.8	48.39	383.6	22.40	37.30
134	25	752.4	4306.6	23.8	42.00	145.3	22.40	37.30
135	18	752.4	4306.6	48.5	36.99	240.8	22.40	37.30
136	19	752.4	4306.6	48.5	36.99	240.8	22.40	37.30
137	17	752.4	4306.6	48.5	25.86	195.9	22.40	37.30
138	27	752.4	4306.6	18.4	13.20	116.4	22.40	37.30
139	24	752.4	4306.6	44.2	12.66	50.0	22.40	37.30
140	21	752.4	4306.6	33.5	5.24	311.2	22.40	37.30
141	23	752.4	4306.6	33.5	4.63	311.2	22.40	37.30
142	22	752.4	4306.6	33.5	4.55	311.2	22.40	37.30
143	42	746.0	4307.8	51.8	5.72	385.0	16.00	37.90
144	87	747.6	4307.8	58.5	242.19	533.3	17.60	37.90
145	80	747.6	4307.8	34.1	149.47	231.3	17.60	37.90
146	86	747.6	4307.8	58.3	82.04	208.5	17.60	37.90
147	79	747.6	4307.8	32.0	56.93	100.0	17.60	37.90
148	85	747.6	4307.8	76.2	51.58	277.4	17.60	37.90
149	78	747.6	4307.8	25.9	7.05	73.8	17.60	37.90
150	88	744.1	4309.7	16.8	3.34	57.8	14.10	38.85

.... TOTAL EMISSION : READ IN= 30286.6, MODEL USED= 14378.0, IN G/SEC

```
... IYR,IMO,IDAY,IHR = 75/ 2/ 1/ 1/ ......
 ***** RAMS DATA *****
```

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
UU	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
DD	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
T1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
T2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CC	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
RA	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

```
RH=  1.00  GRID Z(K)=     0.    20.    40.    60.    80.   100.   125.   150.   175.   200.   225.   250.   275.   300.
                UZF(K)=   1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                VZF(K)=   1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                AKF(K)=   1.92   2.77   4.26   5.51   6.54   7.47   8.26   8.83   9.22   9.45   9.56   9.56   9.47   0.50
             KZ(J=JM/2)=    2.4    2.4    2.5    7.3    4.8    6.6    5.8    3.5    2.4    2.1    1.9    2.0    1.6    1.9    2.2
DT= 120.0     PARM(N)=        5.00        0.0        0.0        0.50     300.00    3545.86   14378.01        0.0        0.0     318.23
  UO=   0.449  PHZ=    0.888  HFZ=    3.901  RIB=    0.0
```

** VERTICAL PROFILE OF * C1 AT I= 15; J= 19**

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Z	K																														
300	14	2	7	9	8	7	12	37	34	59	117	171	133	62	21	51	593	203	234	368	149	47	31	62	108	215	47	3	2	4	6
275	13	2	7	9	8	7	12	37	34	60	117	171	133	62	22	53	607	206	234	367	149	47	31	62	108	214	47	3	2	5	6
250	12	2	8	9	8	7	13	39	36	61	118	172	134	63	23	54	638	213	235	365	149	47	31	62	108	213	47	3	2	5	6
225	11	2	8	10	8	8	14	40	37	62	120	173	135	65	24	55	654	221	236	364	148	48	31	62	107	213	47	3	2	5	7
200	10	2	8	10	9	8	15	43	40	64	122	175	137	67	25	53	614	229	237	362	148	48	31	61	107	212	47	3	2	6	7
175	9	2	8	10	9	8	17	45	43	67	125	178	139	70	28	50	538	235	237	360	146	48	32	61	106	212	47	3	3	7	7
150	8	3	9	11	10	9	19	48	47	71	128	181	143	73	31	45	439	241	238	357	145	48	32	60	105	211	46	3	4	8	8
125	7	3	10	12	11	10	23	52	53	76	133	186	147	78	35	41	343	246	237	354	143	48	32	59	103	210	46	3	4	10	8
100	6	3	10	13	12	11	28	60	61	82	140	192	154	84	40	38	262	248	236	350	140	48	32	59	102	209	45	3	5	11	9
80	5	3	11	13	13	12	34	71	69	88	147	199	160	91	47	38	208	249	235	345	137	49	33	58	101	208	45	3	6	13	9
60	4	4	12	14	14	14	42	87	81	97	156	209	169	100	55	40	164	248	232	340	134	49	34	59	103	207	44	3	8	16	9
40	3	4	12	15	16	16	54	110	94	106	166	222	179	109	66	45	130	246	228	332	129	50	36	60	104	204	44	3	9	18	10
20	2	4	13	16	16	16	53	102	92	105	165	217	178	109	65	42	106	241	221	320	124	49	35	59	101	199	43	3	10	21	10
0	1	4	12	16	16	16	52	101	91	104	163	213	176	109	64	42	105	237	213	307	119	48	34	56	99	193	42	3	10	20	10

YUTM=		4262	4264	4266	4268	4270	4272	4273	4274	4275	4276	4277	4278	4279	4280	4281	4282	4283	4284	4285	4286	4287	4288	4289	4290	4291	4292	4294	4296	4298	4300
	J=	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
Z	K																														
300	14	170	169	159	148	140	132	125	119	112	106	98	86	70	51	34	6	4	5	5	6	6	6	7	7	7	7	7	6	6	5
275	13	171	169	160	148	140	133	126	120	113	107	99	87	71	53	36	7	4	5	5	6	6	6	7	7	7	7	7	6	6	5
250	12	171	169	160	149	141	133	126	120	113	107	99	88	71	54	40	7	4	5	5	6	6	7	7	7	7	7	7	6	6	5
225	11	171	169	160	149	141	133	126	120	112	106	99	88	71	55	45	8	4	5	5	6	6	7	7	7	7	7	7	7	6	6
200	10	170	170	161	149	141	133	125	119	111	105	97	88	70	53	42	8	5	5	5	6	6	7	7	7	7	8	7	7	6	6
175	9	170	170	161	149	140	132	125	117	109	103	95	84	66	50	37	8	5	5	5	6	6	7	7	8	8	8	8	7	7	7
150	8	170	170	161	149	140	132	124	117	107	100	92	80	63	45	30	7	5	5	6	6	7	7	7	8	8	8	8	8	7	8
125	7	169	170	161	149	139	130	123	116	106	98	89	76	59	41	26	7	5	5	6	6	7	7	8	8	8	8	8	8	8	8
100	6	168	170	162	149	138	129	121	115	104	96	87	74	57	38	23	7	6	5	6	6	7	8	8	9	9	9	9	9	9	9
80	5	167	170	162	149	138	127	119	113	103	96	87	74	56	38	23	8	6	6	6	7	7	8	9	9	9	9	9	9	9	10
60	4	165	170	163	149	137	126	117	111	102	96	89	77	57	40	25	10	7	6	6	7	8	8	9	10	10	9	9	10	9	11
40	3	164	170	164	150	138	125	115	110	103	98	93	82	60	45	29	12	9	6	6	7	8	9	10	11	11	10	10	10	10	12
20	2	160	168	163	149	136	122	113	108	101	96	90	78	57	42	27	11	8	6	6	7	8	9	10	11	11	10	10	11	10	12
0	1	156	167	163	149	136	121	112	107	100	95	89	77	57	42	26	11	8	6	6	7	8	9	9	11	11	10	10	11	10	12

*** CC FIELD FOR LAYER K= 1 , RATIO= 1.00
..... PAGE = 1

	XUTM=	725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	0	0	1	1	0	0	0	0	0	0	2	2	4	5	6	0	0	1	2	0	0	0	2	3	0	10	0	0	0	0
4308	39	1	1	1	1	0	0	0	0	0	1	3	3	6	31	27	27	8	26	32	14	0	0	1	4	12	13	1	2	0	0
4306	38	1	1	1	1	0	1	1	1	1	1	0	2	4	25	22	33	9	39	26	12	15	14	7	1	10	7	1	2	0	0
4304	37	1	1	1	1	0	0	1	1	0	0	9	1	3	15	14	21	7	87	18	8	15	25	82	21	18	9	1	1	1	1
4302	36	1	1	1	1	0	1	0	0	1	3	79	4	2	10	9	13	9	129	8	5	10	23	123	20	13	5	0	1	0	0
4300	35	1	0	0	0	0	0	0	0	0	13	165	13	2	7	7	9	12	141	41	3	7	20	121	30	19	7	0	1	2	6
4298	34	0	2	4	3	1	3	0	0	3	25	219	24	2	4	5	7	16	142	141	10	7	19	102	49	135	9	0	1	2	12
4296	33	1	4	6	7	6	5	0	2	3	42	240	38	5	4	5	6	19	139	224	21	2	17	82	54	218	11	0	1	2	10
4294	32	1	3	1	8	8	4	0	4	7	46	223	45	5	2	3	6	21	132	262	32	15	15	66	61	230	14	1	1	2	7
4292	31	3	6	11	8	8	3	1	7	12	48	193	47	9	5	4	5	20	120	255	38	13	12	53	61	209	16	1	1	2	5
4291	30	3	9	18	9	8	5	3	8	14	50	175	47	11	8	5	4	19	111	248	41	12	18	44	64	193	17	1	1	2	5
4290	29	2	11	18	9	8	7	4	8	14	49	156	45	11	8	5	4	18	104	235	41	23	26	51	63	189	17	1	1	2	4
4289	28	3	11	14	9	8	6	3	9	15	44	134	42	12	7	4	3	18	96	229	48	31	28	58	59	178	17	1	1	2	6
4288	27	3	10	14	9	9	7	4	8	15	49	125	40	14	6	4	3	17	94	223	56	36	32	46	55	154	16	1	1	1	8
4287	26	3	11	13	9	8	7	6	8	20	55	118	44	14	5	4	3	20	95	216	71	48	30	37	47	135	15	1	1	1	7
4286	25	4	11	12	9	8	8	7	9	33	52	101	50	15	5	3	4	23	107	196	77	47	23	30	47	117	15	1	0	2	8
4285	24	4	11	12	9	9	10	10	10	39	56	87	46	14	4	2	7	24	124	172	64	37	19	25	46	100	13	1	1	9	10
4284	23	3	11	11	9	8	10	9	17	34	62	83	58	19	4	2	7	45	118	146	53	30	15	20	39	90	13	1	4	13	12
4283	22	3	10	10	9	8	8	18	37	44	66	88	77	36	9	5	9	146	105	119	45	25	13	17	32	78	13	1	9	16	10
4282	21	2	9	11	10	10	13	42	57	65	75	89	88	56	20	8	12	177	91	97	38	21	10	13	30	64	13	2	11	21	8
4281	20	2	9	11	11	12	33	62	70	71	85	93	98	68	39	23	43	177	77	79	33	16	11	12	26	52	12	2	9	18	6
4280	19	2	8	10	11	11	46	90	77	74	93	103	102	81	56	37	88	165	66	61	26	18	16	18	29	42	11	1	8	16	5
4279	18	2	6	9	11	12	37	73	66	77	97	93	103	98	66	49	138	150	55	55	28	27	24	29	48	32	10	5	7	14	4
4278	17	2	7	9	11	18	35	60	53	70	82	76	93	81	65	66	181	138	51	57	41	27	34	33	65	30	7	9	5	11	4
4277	16	2	8	9	12	23	44	59	50	74	81	71	98	71	61	73	210	129	59	65	49	30	40	28	60	24	6	9	4	9	4
4276	15	3	7	8	11	22	36	60	57	72	91	80	95	74	76	74	227	126	71	72	48	35	41	26	57	25	4	10	6	10	5
4275	14	2	7	8	10	27	46	72	61	69	86	93	102	82	92	74	233	115	70	64	48	29	32	32	56	34	11	9	8	12	6
4274	13	5	9	12	13	40	75	92	67	70	101	97	104	95	143	75	236	123	59	50	47	28	28	34	46	34	17	8	11	12	7
4273	12	8	12	15	14	36	81	103	77	73	101	101	113	102	158	74	235	216	55	41	38	32	25	27	38	25	17	6	13	10	6
4272	11	7	12	14	13	31	64	98	87	80	100	115	103	96	157	77	233	266	53	34	30	31	36	21	31	19	20	6	8	10	5
4270	10	4	8	9	12	21	44	30	102	89	106	104	76	80	131	78	215	239	58	34	30	27	35	18	24	15	16	19	9	5	6
4268	9	3	5	6	9	15	35	267	92	55	87	76	57	66	105	76	177	188	49	26	23	20	24	14	18	12	13	18	23	12	8
4266	8	2	4	5	7	13	34	222	76	134	65	55	43	52	84	69	138	142	38	19	16	14	16	11	15	10	9	11	22	29	28
4264	7	1	4	5	7	10	32	135	66	175	53	40	32	41	66	57	103	105	31	14	12	10	12	9	11	8	6	8	14	35	29
4262	6	1	4	5	6	11	27	91	55	137	43	29	24	31	49	44	72	76	24	11	9	7	8	7	8	6	5	6	10	27	19
4260	5	1	3	4	4	10	23	66	45	116	33	20	16	23	32	26	48	53	18	8	6	6	6	5	6	4	4	5	7	19	13
4258	4	1	2	3	4	8	17	46	36	98	26	17	14	16	29	31	29	35	12	5	4	4	4	3	4	3	3	4	6	15	10
4256	3	1	2	2	3	6	13	33	28	76	20	12	9	10	19	21	17	21	8	3	2	3	3	2	3	2	2	3	5	13	8
4254	2	0	1	1	2	4	9	22	21	55	15	7	5	6	11	12	9	12	5	2	1	2	2	2	2	2	2	2	4	10	6
4252	1	0	1	1	2	3	6	15	16	40	11	5	3	4	7	7	5	7	3	1	1	1	1	1	1	1	1	1	3	7	4

** 2HR STATION SO2 ** MO,DAY,HR= 2/ 1/ 2/, NTS= 30, ... SPATIAL AVERAGE... CAL= 50 OBS= 0

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
XS	14.2	12.5	17.6	17.3	13.7	8.7	10.2	18.4	25.4	17.2	8.8	4.5	7.7	14.3	26.1	28.9	27.8	13.1	2.4	-1.0	3.7	11.6	36.2	19.3	-13.8
YS	17.9	24.0	20.5	15.3	14.5	15.6	20.6	29.1	17.9	10.8	10.5	18.9	27.8	32.7	32.9	28.1	10.8	5.6	9.3	23.9	35.2	48.6	24.4	-7.7	20.2
CAL	51	15	91	75	77	76	84	231	13	111	87	12	10	5	5	3	10	44	9	4	1	3	7	1	2
OBS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

```
... IYR,IMO,IDAY,IHR = 75/ 2/ 1/ 2/ ......
 ***** RAMS DATA *****
IS    1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19    20    21    22    23    24    25
UU   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0   5.0
DD   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.   15.
T1    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
T2    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
CC    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
RA    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.

RH=  1.00  GRID Z(K)=    0.    20.    40.    60.    80.   100.   125.   150.   175.   200.   225.   250.   275.   300.
                UZF(K)=  1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                VZF(K)=  1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                AKF(K)=  1.92   2.77   4.26   5.51   6.54   7.47   8.26   8.83   9.22   9.45   9.56   9.56   9.47   0.50
             KZ(J=JM/2)=  2.4    2.4    2.5    7.3    4.8    6.6    5.8    3.5    2.4    2.1    1.9    2.0    1.6    1.9    2.2
DT= 120.0     PARM(N)=      5.00       15.00        0.0         0.50      300.00    3545.86   14378.01        0.0         0.0        318.23
  U0=    0.449  PHZ=    0.888  HFZ=    3.901  RIB=    0.0

** VERTICAL PROFILE OF * C1 AT I= 15; J= 19**

    XUTM= 725 727 729 731 733 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 757 759 761 763
      I=   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30
  Z   K
300  14    9  13  20  46 104 160 124  68  75 122 172 214 242 259 356 605  51  88 135 103  53   6   1   2   0   1   2   4   5   0
275  13    9  14  20  46 104 161 125  68  76 122 172 214 242 259 361 615  53  87 135 103  53   7   1   2   0   1   2   4   5   0
250  12    9  14  20  46 104 161 126  70  77 123 173 215 243 260 366 642  55  86 135 104  53   7   1   2   1   1   2   4   5   0
225  11    9  14  20  46 104 162 129  72  78 125 174 216 244 260 366 660  60  85 135 104  53   7   2   2   1   1   3   5   5   0
200  10   10  14  20  46 104 164 132  74  81 127 176 218 246 260 356 635  69  84 136 105  54   8   3   2   1   1   3   5   6   0
175   9   10  15  21  46 104 167 135  78  84 130 179 221 248 260 341 577  84  82 136 106  54   8   4   2   1   1   3   6   7   0
150   8   10  15  21  46 104 170 140  83  88 134 183 224 251 260 320 501 102  80 137 107  55   9   6   2   1   1   3   7   8   0
125   7   11  16  21  47 104 174 146  90  93 139 188 229 255 261 298 427 120  78 138 108  56  11  10   3   1   2   3   8   9   0
100   6   11  16  22  47 105 180 156  98  99 146 195 235 261 263 279 363 135  76 138 109  58  13  16   4   2   2   4   9  11   0
 80   5   11  17  22  47 105 188 169 108 106 153 202 242 267 267 267 321 145  75 139 110  59  15  19   4   2   2   4  11  13   0
 60   4   12  18  23  48 106 198 187 120 115 163 213 251 276 273 258 287 153  74 139 111  61  17  18   6   2   3   4  13  15   0
 40   3   12  18  24  48 107 211 211 133 125 174 226 262 285 281 254 261 159  73 138 113  65  20  18   8   2   4   4  15  17   0
 20   2   12  18  23  47 105 210 207 134 124 173 223 261 285 280 248 244 161  71 138 114  65  21  16   9   3   5   4  16  19   0
  0   1   12  18  23  45 102 209 210 136 124 171 220 259 284 280 250 246 160  68 136 115  66  22  16   9   2   5   4  16  19   0

    YUTM=426242644266426842704272427342744275427642774278427942804281428242834284428542864287428842894290429142924294429642984300
      J=    6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35
  Z   K
300  14   137 146 163 197 255 343 413 492 621 561 569 640 455 356 288 181 185 220 250 278 302 320 330 331 326 312 271 191 105  35
275  13   137 146 164 198 255 343 411 485 576 546 567 617 449 361 295 184 184 220 250 278 301 319 328 330 324 311 270 190 105  35
250  12   138 147 165 199 256 342 406 472 539 533 563 606 446 366 313 191 183 219 249 276 299 317 326 327 321 308 266 187 104  37
225  11   139 149 167 201 258 341 399 457 510 520 557 614 446 366 330 199 182 218 247 274 296 313 322 323 316 303 261 182 102  39
200  10   141 151 170 205 261 340 392 443 487 506 545 616 440 356 319 197 183 217 245 271 293 308 317 317 310 296 253 176  99  41
175   9   143 154 173 208 264 339 386 431 469 492 522 555 414 341 297 193 185 216 242 267 288 303 310 310 303 288 244 168  95  42
150   8   145 157 177 213 268 339 380 420 454 476 496 498 387 320 269 188 187 214 240 263 283 297 304 303 295 279 234 159  90  43
125   7   148 161 181 218 273 340 376 411 441 461 471 455 362 298 246 185 189 213 237 259 278 291 297 295 286 269 224 150  85  42
100   6   151 165 186 223 277 340 372 404 431 447 449 422 341 279 231 183 191 211 234 255 272 284 289 287 277 260 213 141  80  42
 80   5   154 169 191 228 282 341 370 400 425 438 435 402 327 267 222 183 191 211 233 252 268 279 284 281 271 253 205 134  76  41
 60   4   158 174 196 234 287 342 369 397 421 431 424 388 316 258 216 183 192 210 231 249 264 274 278 275 264 246 198 128  72  40
 40   3   162 179 202 240 293 345 369 397 419 427 419 380 310 254 215 185 194 210 229 246 260 270 273 270 259 240 191 122  69  39
 20   2   166 185 206 245 298 347 371 399 419 425 412 370 303 248 211 185 194 209 227 243 256 265 268 265 254 235 186 118  66  38
  0   1   169 189 210 249 302 351 376 404 424 431 417 373 305 250 212 186 195 209 227 242 254 263 266 263 252 233 185 117  65  38
```

*** CC FIELD FOR LAYER K= 1 , RATIO= 1.00
..... PAGE = 1

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	0	0	1	1	0	0	0	0	0	1	3	2	4	5	4	0	0	2	2	1	0	1	3	2	2	9	0	0	0	0
4308	39	1	1	1	1	0	1	1	1	1	2	4	4	13	36	29	27	10	28	33	7	0	0	1	8	15	12	1	2	0	0
4306	38	1	1	1	1	0	1	1	1	0	0	4	5	15	33	23	24	10	67	19	8	12	9	15	3	23	5	1	2	0	0
4304	37	1	1	1	1	1	1	0	0	2	13	2	6	14	21	16	33	63	57	5	8	29	67	56	20	17	6	0	2	0	0
4302	36	1	1	1	1	0	1	1	8	42	57	10	7	12	17	22	55	87	32	3	19	54	86	39	7	15	2	0	1	0	0
4300	35	1	0	0	0	0	1	11	48	98	79	31	7	11	18	33	75	103	49	8	31	64	75	61	73	15	5	0	1	3	5
4298	34	1	3	5	3	2	10	37	92	133	83	54	7	11	23	53	111	148	99	24	41	66	83	127	147	18	3	0	1	4	12
4296	33	2	6	8	8	9	26	70	128	136	90	72	10	17	39	87	158	190	127	48	46	66	107	162	125	35	1	0	2	4	9
4294	32	3	4	2	11	15	40	93	137	126	91	82	14	26	62	123	190	199	134	74	56	91	123	159	95	50	1	1	2	4	6
4292	31	5	9	14	12	20	51	104	133	117	98	90	23	43	88	146	195	188	137	95	63	96	125	141	78	57	2	1	3	4	4
4291	30	6	12	21	13	23	59	108	128	115	103	93	28	52	99	154	192	180	139	105	67	99	132	126	76	58	2	1	3	4	4
4290	29	6	15	22	14	25	65	107	122	111	107	94	33	60	106	157	186	172	142	113	70	115	137	128	74	63	2	2	3	4	3
4289	28	8	16	18	15	28	67	104	117	110	106	94	39	67	108	155	180	167	145	121	78	124	136	130	71	74	2	2	3	4	5
4288	27	8	16	18	16	30	69	103	112	111	115	99	44	73	110	153	174	161	152	135	88	129	134	113	71	67	2	2	3	4	7
4287	26	8	17	18	16	30	69	101	110	118	129	114	54	77	110	150	168	167	165	156	105	139	123	101	69	63	2	1	2	3	7
4286	25	10	17	17	17	32	71	101	113	133	131	119	64	78	110	148	169	180	185	173	113	132	110	93	73	60	2	1	2	5	7
4285	24	10	18	17	17	33	72	102	114	144	141	119	63	78	109	146	169	186	208	169	102	119	100	90	77	56	2	1	4	13	9
4284	23	10	18	16	17	32	71	97	124	144	154	127	78	84	113	144	169	215	216	166	93	109	94	87	74	55	2	2	7	17	11
4283	22	10	17	16	18	32	69	114	146	156	165	144	101	101	117	147	193	317	201	154	88	102	89	85	71	57	3	3	13	20	8
4282	21	9	17	16	19	34	80	143	168	179	175	153	118	119	127	153	225	343	177	146	84	97	84	83	72	54	4	4	16	24	5
4281	20	10	17	17	20	38	106	166	182	187	188	161	133	131	150	194	278	326	156	140	81	90	84	79	71	51	5	4	15	21	3
4280	19	10	16	16	21	40	125	197	189	193	197	173	142	149	178	235	327	288	142	132	76	93	90	95	71	48	6	3	15	20	2
4279	18	10	15	16	22	44	119	181	180	199	199	163	149	169	207	293	363	242	133	131	83	103	107	132	75	44	6	9	14	18	2
4278	17	10	16	16	23	51	121	168	170	191	178	145	143	160	227	356	369	201	133	141	100	106	123	132	72	43	6	13	12	14	2
4277	16	11	17	17	26	60	131	167	167	193	172	136	143	169	279	388	351	170	147	152	109	111	129	114	63	41	6	13	10	12	2
4276	15	11	16	17	26	61	125	175	169	189	177	145	157	191	328	391	325	155	164	159	108	117	126	101	64	40	5	14	12	13	3
4275	14	11	16	17	28	71	144	187	166	183	171	159	178	246	346	380	293	139	169	149	108	110	115	100	71	50	11	14	14	15	4
4274	13	14	19	22	33	90	177	202	166	182	184	168	217	338	337	363	267	149	158	131	107	107	109	97	65	51	17	12	17	15	4
4273	12	17	23	26	37	90	186	209	172	182	184	183	253	352	319	342	275	244	137	115	98	110	100	84	59	42	18	11	20	13	3
4272	11	17	23	26	38	90	179	206	177	188	186	210	263	339	309	330	297	281	102	101	91	109	105	72	53	34	20	10	15	12	3
4270	10	14	20	24	40	82	133	116	224	180	201	213	251	305	297	313	304	236	78	93	89	102	94	58	44	25	19	24	14	9	4
4268	9	13	17	22	40	79	169	390	123	190	188	197	237	274	283	294	277	178	58	81	79	87	74	48	35	18	17	25	29	14	6
4266	8	12	17	23	41	81	184	351	199	166	167	182	219	248	268	271	240	131	46	70	68	73	59	40	28	14	14	20	29	35	24
4264	7	12	17	24	41	84	186	292	223	142	155	169	200	228	251	246	205	100	41	61	59	62	48	32	21	11	12	17	27	41	22
4262	6	11	18	24	44	86	181	241	174	133	144	157	185	214	234	220	173	79	39	53	51	52	39	26	16	9	12	17	26	31	12
4260	5	11	17	24	45	87	171	203	153	125	130	144	172	198	208	188	145	65	37	46	44	43	32	21	13	9	12	17	24	23	6
4258	4	11	16	23	44	87	157	171	134	119	119	139	165	186	199	174	116	54	33	38	36	35	26	17	11	9	12	16	22	19	4
4256	3	10	15	23	44	85	143	144	116	114	109	130	153	173	177	145	90	45	28	31	29	28	21	14	11	10	12	16	20	16	3
4254	2	10	14	22	43	82	128	122	104	108	99	121	141	158	151	114	69	36	24	25	23	23	17	12	10	10	12	15	18	14	3
4252	1	9	13	22	42	77	114	106	94	100	91	112	130	141	128	91	53	29	20	21	19	18	14	10	9	9	11	14	16	12	2

** 2HR STATION SO2 ** MO,DAY,HR= 2/ 1/ 3/, NTS= 60, ... SPATIAL AVERAGE... CAL= 115 OBS= 0

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
XS	14.2	12.5	17.6	17.3	13.7	8.7	10.2	18.4	25.4	17.2	8.8	4.5	7.7	14.3	26.1	28.9	27.8	13.1	2.4	-1.0	3.7	11.6	36.2	19.3	-13.8
YS	17.9	24.0	20.5	15.3	14.5	15.6	20.6	29.1	17.9	10.8	10.5	18.9	27.8	32.7	32.9	28.1	10.8	5.6	9.3	23.9	35.2	48.6	24.4	-7.7	20.2
CAL	249	79	174	154	342	190	159	112	11	159	187	39	117	72	1	4	16	228	23	10	1	3	7	18	9
OBS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

... IYR,IMO,IDAY,IHR = 75/ 2/ 1/ 3/ ......

***** RAMS DATA *****

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
UU	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
DD	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.	30.
T1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
T2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CC	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
RA	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

```
RH=  1.00  GRID Z(K)=     0.    20.    40.    60.    80.   100.   125.   150.   175.   200.   225.   250.   275.   300.
                UZF(K)=   1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                VZF(K)=   1.00   1.11   1.23   1.31   1.37   1.41   1.46   1.49   1.49   1.49   1.49   1.49   1.49   1.49
                AKF(K)=   1.92   2.77   4.26   5.51   6.54   7.47   8.26   8.83   9.22   9.45   9.56   9.56   9.47   0.50
            KZ(J=JM/2)=    2.4    2.4    2.5    7.3    4.8    6.6    5.8    3.5    2.4    2.1    1.9    2.0    1.6    1.9    2.2
DT= 150.0    PARM(N)=        5.00       30.00        0.0        0.50     300.00   3545.86  14378.01       0.0        0.0        318.23
  U0=    0.449  PHZ=    0.888  HFZ=    3.901  RIB=    0.0
```

** VERTICAL PROFILE OF * C1 AT I= 15; J= 19**

	XUTM=	725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Z	K																														
300	14	76	99	101	79	103	217	217	162	119	96	101	134	194	310	440	206	8	1	2	3	3	3	2	1	2	2	3	3	1	0
275	13	76	99	101	79	103	218	218	162	119	96	101	135	194	311	443	211	8	1	2	3	3	3	2	1	2	2	3	3	1	0
250	12	76	99	101	79	104	219	219	163	121	97	102	136	195	311	452	222	9	1	2	3	3	3	3	2	2	2	3	4	2	0
225	11	76	100	101	80	104	220	221	165	122	99	104	137	196	311	455	230	9	2	2	3	3	3	3	2	2	2	3	4	2	0
200	10	76	100	101	80	104	222	224	168	125	102	106	140	198	308	439	219	9	2	2	3	3	4	3	2	2	2	3	5	2	0
175	9	76	100	102	81	105	225	228	172	128	105	109	142	200	303	410	194	9	2	2	3	3	4	4	2	2	3	4	6	3	0
150	8	76	100	102	81	106	228	234	177	133	109	113	146	202	297	372	162	8	3	2	3	4	6	6	2	2	3	4	7	3	0
125	7	76	100	103	82	106	233	241	184	139	115	119	151	206	291	334	133	7	3	3	3	4	7	8	3	2	3	5	8	4	0
100	6	75	100	104	84	108	240	252	193	146	122	126	158	210	286	301	108	7	4	3	3	5	10	11	3	3	4	6	9	5	0
80	5	75	100	105	85	110	248	265	202	154	130	134	165	215	284	279	93	7	5	3	4	6	12	14	4	3	5	6	11	6	0
60	4	75	101	106	87	112	260	284	215	164	140	145	174	223	284	262	81	7	5	3	4	7	13	15	5	3	5	7	13	7	0
40	3	75	101	107	89	115	275	309	229	175	151	159	184	230	287	250	72	7	6	3	5	9	16	16	6	4	6	8	15	9	0
20	2	74	100	107	89	113	272	303	229	177	152	157	184	231	283	240	67	6	6	3	5	9	16	16	7	4	7	8	17	10	0
0	1	72	98	107	89	111	269	304	231	178	152	156	183	230	283	241	69	6	6	3	4	9	16	16	7	4	7	8	17	10	0

	YUTM=	4262	4264	4266	4268	4270	4272	4273	4274	4275	4276	4277	4278	4279	4280	4281	4282	4283	4284	4285	4286	4287	4288	4289	4290	4291	4292	4294	4296	4298	4300
	J=	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
Z	K																														
300	14	14	16	20	33	52	52	20	17	97	47	138	340	341	440	407	147	109	102	110	115	113	103	89	77	73	82	138	302	487	457
275	13	14	16	21	33	53	54	21	15	70	45	137	325	339	443	424	152	111	102	110	115	113	103	89	77	73	82	137	299	481	455
250	12	14	16	21	34	55	60	26	13	48	44	136	319	337	452	471	164	117	103	110	115	112	103	89	77	72	81	134	289	465	446
225	11	14	16	21	35	59	69	34	12	33	42	135	324	333	455	514	181	128	104	110	115	112	103	89	77	72	79	128	275	439	427
200	10	14	16	22	37	64	83	48	12	24	40	131	324	324	439	495	193	146	106	111	115	112	103	89	77	71	77	121	256	406	395
175	9	15	17	23	39	70	101	72	14	18	38	122	284	307	410	444	207	175	110	111	115	112	103	89	77	70	74	113	235	366	355
150	8	15	17	23	41	77	121	109	17	14	35	112	246	287	372	382	221	218	116	111	115	112	103	89	76	69	72	105	212	325	313
125	7	15	18	24	44	84	141	165	22	11	33	103	216	266	334	334	235	276	125	111	115	112	102	89	76	68	69	96	190	285	272
100	6	15	18	26	47	91	154	184	25	10	30	95	193	247	301	300	238	273	126	112	115	111	102	89	75	66	66	87	168	248	234
80	5	15	19	27	49	95	155	161	23	9	29	90	180	234	279	279	233	238	121	115	116	111	102	88	75	65	64	81	152	221	207
60	4	16	19	28	52	100	152	122	18	9	28	87	170	223	262	262	224	208	118	117	116	111	101	88	74	65	62	75	138	198	183
40	3	16	20	29	[illegible]	105	146	93	[illegible]	8	27	86	165	215	250	251	216	187	116	118	116	110	101	87	74	64	60	70	126	178	162
20	2	16	20	30	56	106	139	73	11	8	27	84	158	208	240	241	208	173	114	117	115	110	100	87	73	63	59	66	117	163	148
0	1	16	20	29	56	106	138	72	11	8	29	86	159	209	241	241	208	173	114	117	115	109	99	86	73	63	58	66	116	161	146

*** CC FIELD FOR LAYER K= 1 , RATIO= 1.00
..... PAGE = 1

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	0	0	1	0	0	0	0	0	0	1	2	2	3	5	3	0	0	2	3	1	0	1	2	2	5	7	0	0	0	0
4308	39	1	1	1	1	0	1	1	1	1	2	4	9	25	36	25	20	15	34	23	2	0	0	3	13	15	8	1	2	0	0
4306	38	1	1	1	1	1	1	0	0	0	2	9	15	25	26	19	27	37	47	9	6	12	15	18	9	16	1	1	2	0	0
4304	37	1	1	1	1	1	0	0	3	19	17	8	12	13	15	28	51	50	23	6	28	59	68	39	17	8	2	0	1	0	0
4302	36	1	1	1	0	0	1	27	65	62	18	4	9	17	41	77	89	49	12	24	49	59	41	17	12	6	0	0	0	0	0
4300	35	1	0	0	0	2	42	103	106	55	10	6	22	54	98	127	106	44	15	37	57	72	82	64	17	4	2	0	1	3	5
4298	34	1	3	4	5	32	112	132	92	36	9	25	64	110	149	147	100	43	32	61	95	125	130	79	20	2	2	0	2	5	9
4296	33	3	7	10	26	78	148	118	68	29	34	71	120	161	173	141	91	56	60	96	129	136	108	48	13	1	1	1	3	5	6
4294	32	4	6	14	52	109	142	91	58	43	68	113	153	173	163	128	94	77	89	119	135	116	66	32	8	0	1	1	3	4	3
4292	31	8	16	37	70	119	125	79	64	67	99	139	165	172	154	123	99	88	98	116	112	81	44	23	5	0	2	2	3	4	1
4291	30	10	23	50	78	121	118	76	69	80	114	148	167	169	150	122	101	91	100	111	98	66	39	18	6	0	2	2	4	3	1
4290	29	13	30	56	84	122	111	75	74	90	123	152	167	164	145	121	101	92	99	103	88	67	44	20	9	5	2	3	4	3	1
4289	28	17	34	57	89	121	102	74	80	99	127	153	165	159	140	120	100	91	96	97	83	69	44	29	9	14	1	3	4	3	2
4288	27	19	36	61	93	119	97	75	84	109	143	166	163	156	137	117	96	87	96	97	82	65	40	16	8	7	0	3	3	3	4
4287	26	22	40	65	96	115	92	78	92	128	165	178	162	152	134	112	93	92	103	105	89	68	29	8	4	4	1	3	2	2	3
4286	25	25	44	68	98	112	89	83	109	151	172	179	161	146	129	108	97	107	113	113	93	55	17	6	6	2	1	2	2	4	4
4285	24	27	46	71	100	110	89	91	124	168	183	174	153	142	125	106	102	120	121	97	73	37	11	7	8	0	1	2	5	11	5
4284	23	29	49	73	99	106	87	100	142	177	194	175	153	139	118	105	113	127	118	85	56	26	9	5	5	2	1	3	9	16	6
4283	22	30	50	75	100	102	90	128	171	193	203	189	170	145	126	166	177	120	101	69	43	20	7	4	4	6	3	5	15	18	3
4282	21	32	52	78	101	103	110	172	200	215	211	199	183	164	164	206	172	104	84	57	35	15	4	4	6	4	4	8	20	21	0
4281	20	34	54	80	102	109	148	198	216	225	223	207	197	189	215	250	163	87	70	50	28	11	5	5	7	3	6	9	18	15	0
4280	19	36	55	81	102	117	186	234	223	231	233	225	215	231	278	265	143	72	60	43	23	13	14	16	7	3	7	8	17	11	0
4279	18	37	56	82	103	124	187	215	219	242	238	227	244	288	318	258	115	58	54	41	27	28	43	48	12	2	8	13	15	9	0
4278	17	39	59	84	107	136	186	202	214	229	221	225	262	307	331	239	87	50	59	57	41	42	58	42	8	5	9	18	11	5	0
4277	16	42	63	86	112	149	199	203	215	233	223	240	291	337	336	192	65	54	75	70	52	48	55	21	3	9	9	16	8	4	0
4276	15	43	63	88	115	154	199	214	218	236	251	277	332	361	312	138	55	68	95	80	54	56	44	13	8	11	9	18	8	5	1
4275	14	45	66	90	119	165	217	225	217	241	268	321	371	368	257	100	54	81	100	77	54	42	31	20	22	24	14	16	11	8	2
4274	13	50	71	96	130	195	255	238	222	262	309	357	428	394	185	85	66	83	86	67	55	36	28	24	21	29	21	14	13	8	2
4273	12	55	76	103	136	200	270	249	238	282	337	399	446	330	140	141	148	77	77	60	49	36	22	18	19	24	20	12	15	5	1
4272	11	56	77	104	142	205	258	243	260	316	382	432	387	251	165	188	127	65	66	53	44	39	27	12	18	22	22	11	10	5	1
4270	10	55	77	107	149	194	213	235	323	352	392	365	289	216	179	148	89	69	62	55	44	34	23	11	18	20	22	22	8	2	2
4268	9	56	78	110	153	203	316	432	273	363	336	296	242	194	147	101	68	63	54	50	35	23	14	13	18	19	21	24	20	7	5
4266	8	57	81	117	162	224	356	433	322	306	287	254	210	162	113	76	58	54	47	41	26	16	11	14	17	17	17	19	25	30	18
4264	7	60	86	124	173	250	366	380	306	258	257	220	178	133	94	66	53	48	41	34	21	13	12	14	16	15	15	17	26	35	13
4262	6	63	92	133	190	266	346	310	258	239	229	190	151	113	82	60	48	43	37	30	19	13	12	13	14	14	15	18	24	21	3
4260	5	66	97	142	201	270	316	272	240	216	199	162	128	95	70	53	45	39	34	27	18	13	11	12	13	14	15	18	19	9	0
4258	4	70	101	150	208	265	287	244	217	193	176	145	116	90	74	56	38	36	32	25	17	13	11	12	13	15	16	17	14	4	0
4256	3	74	107	156	210	255	258	215	193	174	156	130	106	82	64	46	34	34	29	23	17	13	11	12	14	15	16	15	10	2	0
4254	2	78	112	160	208	240	228	188	172	157	139	116	94	70	50	37	33	21	26	20	15	12	11	12	14	15	15	13	7	1	0
4252	1	82	115	160	201	221	200	167	156	142	125	104	82	58	41	33	30	28	23	18	14	11	11	12	14	14	13	10	4	0	0

** 2HR STATION SO2 ** MO,DAY,HR= 2/ 1/ 4/, NTS= 84, ... SPATIAL AVERAGE... CAL= 118 OBS= 0

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
XS	14.2	12.5	17.6	17.3	13.7	8.7	10.2	18.4	25.4	17.2	8.8	4.5	7.7	14.3	26.1	28.9	27.8	13.1	2.4	-1.0	3.7	11.6	36.2	19.3	-13.8
YS	17.9	24.0	20.5	15.3	14.5	15.6	20.6	29.1	17.9	10.8	10.5	18.9	27.8	32.7	32.9	28.1	10.8	5.6	9.3	23.9	35.2	48.6	24.4	-7.7	20.2
CAL	275	145	82	75	275	232	202	107	8	70	338	114	83	147	1	3	10	98	104	26	1	2	4	13	31
OBS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

... IYR,IMO,IDAY,IHR = 75/ 2/ 1/ 4/ ......

***** RAMS DATA *****

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
UU	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
DD	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.	45.
T1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
T2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CC	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
RA	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

RH= 1.00

GRID Z(K)=	0.	20.	40.	60.	80.	100.	125.	150.	175.	200.	225.	250.	275.	300.	
UZF(K)=	1.00	1.11	1.23	1.31	1.37	1.41	1.46	1.49	1.49	1.49	1.49	1.49	1.49	1.49	
VZF(K)=	1.00	1.11	1.23	1.31	1.37	1.41	1.46	1.49	1.49	1.49	1.49	1.49	1.49	1.49	
AKF(K)=	1.92	2.77	4.26	5.51	6.54	7.47	8.26	8.83	9.22	9.45	9.56	9.56	9.47	0.50	
KZ(J=JM/2)=	2.4	2.4	2.5	7.3	4.8	6.6	5.8	3.5	2.4	2.1	1.9	2.0	1.6	1.9	2.2

DT= 180.0 PARM(N)= 5.00 45.00 0.0 0.50 300.00 3545.86 14378.01 0.0 0.0 318.23

U0= 0.449 PHZ= 0.888 HFZ= 3.901 RIB= 0.0

** VERTICAL PROFILE OF * C1 AT I= 15; J= 19**

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Z	K																														
300	14	125	139	138	121	119	109	59	42	49	77	141	274	355	195	15	3	3	2	1	1	2	2	1	1	2	3	4	3	0	0
275	13	125	139	138	121	120	109	59	42	49	78	142	274	354	197	15	3	3	2	1	1	2	2	1	1	2	3	4	3	0	0
250	12	125	139	138	121	120	111	60	43	50	79	143	274	351	202	16	3	3	2	1	1	2	3	2	2	2	3	4	3	0	0
225	11	125	139	139	122	122	113	63	45	52	81	145	274	347	205	17	3	3	2	2	1	2	3	2	2	2	3	4	3	0	0
200	10	125	140	139	123	123	116	66	48	55	83	147	273	342	199	16	4	3	2	2	1	2	4	2	2	3	4	5	4	0	0
175	9	125	140	140	124	125	120	70	52	59	87	150	272	335	186	16	4	3	3	2	1	3	6	3	2	3	4	6	4	0	0
150	8	125	140	141	125	128	125	75	57	63	91	154	271	326	171	14	4	3	3	2	2	3	8	4	2	3	5	6	5	0	0
125	7	125	140	142	127	131	133	82	63	69	97	159	271	318	157	14	4	4	3	2	2	4	12	5	3	4	5	7	6	0	0
100	6	125	141	143	129	135	143	93	72	77	104	167	272	311	146	13	5	4	3	2	2	6	17	7	3	5	6	8	7	1	0
80	5	125	141	144	132	140	154	104	81	85	112	175	275	307	140	13	5	5	3	2	3	7	20	9	4	5	7	9	9	1	0
60	4	125	141	146	134	145	170	121	93	96	122	186	282	306	137	12	6	5	4	3	4	9	21	10	5	6	8	10	10	2	0
40	3	125	141	147	138	152	190	141	107	107	133	199	289	306	136	12	7	6	4	3	5	12	23	13	5	7	9	11	12	3	0
20	2	124	141	148	139	153	190	137	107	109	134	199	288	302	132	12	6	6	4	3	5	12	22	13	6	7	9	11	14	4	0
0	1	123	140	148	140	153	190	138	108	109	134	198	286	300	131	12	6	5	4	3	5	12	22	13	6	7	9	11	14	4	0

YUTM=426242644266426842704272427342744275427642774278427942804281428242834284428542864287428842894290429142924294429642984300

	J=	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
Z	K																														
300	14	8	11	14	16	19	43	64	27	25	5	3	4	5	15	103	54	68	21	8	3	4	12	27	49	72	93	118	107	235	657
275	13	8	11	14	16	19	44	67	28	21	5	3	4	5	15	107	55	72	22	8	3	4	12	27	49	72	93	118	107	227	627
250	12	9	11	14	16	19	46	76	30	18	5	3	4	5	16	120	61	83	25	8	3	4	12	27	49	72	93	118	106	212	572
225	11	9	11	14	16	20	51	94	33	17	5	3	4	5	17	131	68	105	29	9	3	4	12	27	49	72	92	118	105	191	500
200	10	9	12	15	17	21	57	124	39	16	5	3	4	5	16	125	75	138	36	10	4	4	12	27	49	72	92	118	104	167	419
175	9	9	12	15	18	22	65	171	49	16	6	4	5	5	16	109	84	191	47	12	4	4	12	27	49	72	92	118	103	142	337
150	8	9	12	16	19	24	75	245	63	17	6	4	5	5	14	90	94	272	63	14	5	4	12	27	48	72	92	117	101	120	264
125	7	9	13	17	20	26	87	351	83	19	7	4	5	6	14	75	105	376	86	17	6	4	12	27	48	71	91	117	100	101	204
100	6	10	13	18	22	29	95	386	95	20	8	5	6	6	13	64	110	368	89	20	7	5	12	27	48	71	91	116	98	86	157
80	5	10	14	18	24	32	97	343	91	22	9	6	7	6	13	58	109	300	78	23	8	5	12	27	48	71	90	116	97	77	127
60	4	10	14	19	26	35	96	271	79	24	11	8	9	7	12	54	106	241	69	26	10	5	12	27	48	70	90	115	96	69	104
40	3	11	15	20	28	39	95	218	70	25	12	10	11	8	12	50	104	200	64	28	11	5	13	27	48	70	89	115	95	63	86
20	2	11	15	21	29	41	92	182	64	27	13	9	10	7	12	48	101	172	59	29	11	5	13	27	48	70	89	114	95	58	73
0	1	11	15	21	29	41	91	180	63	27	13	9	10	7	12	47	101	170	59	29	11	5	13	27	47	70	89	113	94	58	72

*** CC FIELD FOR LAYER K= 1 , RATIO= 1.00
..... PAGE = 1

XUTM=		725	727	729	731	733	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	757	759	761	763
	I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
4310	40	1	1	1	0	0	0	0	0	0	1	2	2	5	5	0	0	2	5	3	1	1	2	3	4	8	5	0	0	0	0
4308	39	1	1	1	1	1	1	1	1	3	6	10	18	34	29	28	22	32	32	13	0	0	3	9	16	11	5	1	2	0	0
4306	38	1	1	1	1	0	0	0	3	7	9	13	17	23	27	39	42	25	15	11	25	39	27	11	12	7	1	1	2	0	0
4304	37	1	1	1	0	0	13	28	21	8	6	8	15	29	47	54	33	27	34	51	61	49	29	21	11	4	2	0	1	0	0
4302	36	1	1	0	4	37	78	52	23	7	8	26	54	82	97	84	53	36	40	39	28	18	19	20	10	2	0	0	0	0	0
4300	35	0	0	9	47	91	86	37	17	25	58	97	131	142	127	85	46	33	42	56	73	86	65	27	6	4	2	0	1	4	4
4298	34	4	19	54	95	109	65	42	61	97	137	167	173	148	106	67	55	71	98	120	124	98	48	12	2	2	2	1	3	7	7
4296	33	22	49	88	117	104	75	95	129	163	187	185	161	123	91	80	93	113	124	114	83	43	13	2	1	1	1	2	4	5	3
4294	32	39	68	100	118	104	109	144	171	183	177	157	129	105	96	100	109	109	96	70	43	16	3	4	2	1	2	3	4	3	1
4292	31	59	91	118	115	115	138	166	176	171	155	134	114	102	98	97	93	81	60	37	18	6	3	5	3	2	3	3	3	2	0
4291	30	69	103	125	115	123	153	171	173	162	144	124	109	102	97	91	82	65	43	23	10	7	4	6	4	2	3	3	3	2	0
4290	29	79	111	123	117	132	160	168	165	151	132	116	105	97	90	82	68	50	30	16	18	18	16	11	11	6	3	3	3	1	0
4289	28	86	112	119	121	139	161	162	154	136	120	108	99	89	81	71	55	37	23	19	25	24	23	18	14	9	2	3	2	2	1
4288	27	91	113	120	125	144	160	153	145	138	130	104	91	82	72	59	42	29	27	27	30	25	16	11	8	3	2	2	2	2	3
4287	26	95	115	120	129	145	155	149	152	153	129	98	82	73	62	49	39	36	42	42	40	23	9	5	3	2	2	2	1	2	2
4286	25	99	116	121	132	146	154	159	170	143	109	89	75	64	55	47	51	57	51	49	33	11	4	6	4	1	2	1	1	4	3
4285	24	101	117	122	134	148	161	167	159	132	93	76	66	58	53	54	64	67	46	26	12	4	5	9	4	1	1	2	6	12	3
4284	23	103	117	123	135	149	163	163	140	122	98	76	64	54	52	74	72	58	32	12	5	4	5	4	3	2	2	4	11	16	3
4283	22	103	117	124	137	150	168	163	142	120	105	93	80	71	113	178	73	37	14	4	4	5	4	3	5	5	4	8	18	14	0
4282	21	104	118	127	140	157	184	181	153	132	113	107	110	140	203	113	37	18	6	4	4	3	3	6	5	4	6	11	22	13	0
4281	20	105	119	130	145	172	208	177	152	140	131	140	182	259	265	64	18	8	5	5	3	3	6	7	5	6	8	13	17	7	0
4280	19	106	119	131	150	135	228	188	159	155	168	215	291	310	156	31	10	6	4	4	6	12	22	14	6	7	9	11	14	5	0
4279	18	107	119	134	157	187	195	161	169	187	221	294	320	225	78	21	6	5	7	12	23	41	60	31	9	7	11	14	10	3	0
4278	17	108	122	138	165	189	167	154	182	212	264	299	247	148	60	17	5	10	27	40	44	64	47	20	10	10	13	17	4	1	0
4277	16	110	126	144	171	188	175	176	216	263	281	265	198	101	40	12	11	34	54	59	58	51	27	8	10	12	12	12	2	1	0
4276	15	112	128	148	172	183	187	212	253	288	285	238	167	89	33	12	31	63	75	65	52	38	18	13	15	12	11	12	3	3	1
4275	14	114	132	152	174	195	224	244	268	282	260	223	183	80	26	23	51	73	71	58	34	19	23	26	30	21	14	10	6	7	2
4274	13	121	140	160	188	239	279	266	270	279	250	216	166	57	22	56	62	60	51	42	29	22	28	27	27	26	18	8	10	6	1
4273	12	127	147	169	199	252	300	279	269	262	238	199	121	42	84	170	75	46	38	29	29	27	22	21	21	20	15	7	12	3	0
4272	11	129	151	175	213	261	285	278	271	253	219	151	97	117	154	95	40	32	29	28	31	34	21	15	17	19	15	6	6	3	1
4270	10	134	158	188	222	243	243	259	245	210	158	117	104	102	76	49	38	35	33	31	29	26	18	15	16	17	19	17	4	1	2
4268	9	142	168	198	231	268	364	288	177	151	110	85	72	59	43	32	29	28	25	22	18	15	14	14	15	16	17	18	16	7	4
4266	8	151	180	213	251	291	328	251	132	97	72	57	46	38	29	23	20	18	16	14	13	12	12	13	13	13	13	18	29	31	13
4264	7	163	196	233	269	283	237	152	87	63	49	39	32	27	22	17	15	13	12	11	11	11	11	12	12	12	14	20	30	29	6
4262	6	179	215	250	264	230	139	84	58	45	36	29	24	19	16	13	12	11	11	10	10	10	11	11	12	14	16	19	19	11	0
4260	5	196	229	249	234	173	90	57	41	32	25	19	14	11	10	11	11	10	10	10	10	10	11	12	13	14	15	14	9	1	0
4258	4	208	232	232	195	128	59	41	31	25	22	19	16	18	21	13	8	10	10	10	10	11	12	12	13	13	12	9	3	0	0
4256	3	212	225	206	157	92	38	32	26	22	19	17	15	15	14	10	8	9	9	10	11	12	12	12	11	10	9	5	2	1	0
4254	2	209	208	176	123	68	28	26	22	19	16	13	11	10	9	8	9	9	10	10	11	11	11	10	9	7	6	3	1	1	0
4252	1	198	188	149	99	53	23	23	20	16	13	11	9	8	8	9	9	10	10	10	10	10	9	8	6	5	4	1	0	0	0

** 2HR STATION SO2 ** MO,DAY,HR= 2/ 1/ 5/, NTS= 104, ... SPATIAL AVERAGE... CAL= 77 OBS= 0

IS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
XS	14.2	12.5	17.6	17.3	13.7	8.7	10.2	18.4	25.4	17.2	8.8	4.5	7.7	14.3	26.1	28.9	27.8	13.1	2.4	-1.0	3.7	11.6	36.2	19.3	-13.8
YS	17.9	24.0	20.5	15.3	14.5	15.6	20.6	29.1	17.9	10.8	10.5	18.9	27.8	32.7	32.9	28.1	10.8	5.6	9.3	23.9	35.2	48.6	24.4	-7.7	20.2
CAL	51	62	7	55	29	264	108	22	11	34	242	179	154	81	2	2	7	19	187	100	26	4	3	10	104
OBS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

5. Report of IBMAQ-1

Reprinted from JOURNAL OF APPLIED METEOROLOGY, Vol. 13, No. 2, March 1974, pp. 185–204
American Meteorological Society
Printed in U. S. A.

A Generalized Urban Air Pollution Model and Its Application to the Study of SO_2 Distributions in the St. Louis Metropolitan Area

C. C. SHIR
IBM Research Laboratory, San Jose, Calif. 95193

L. J. SHIEH
IBM Scientific Center, Palo Alto, Calif. 94304

(Manuscript received 6 June 1973, in revised form 26 December 1973)

ABSTRACT

A generalized urban air pollution model, based on numerical integration of the concentration equation, is developed for the study of air pollutant distributions over an urban area. The model computes the temporal and three-dimensional spatial concentration distributions resulting from specified multiple point and area sources by using currently available meteorological and source inventory data. A new method based on experiments and a turbulence transport model is used to estimate the turbulent diffusivity and atmospheric stability. Special treatments of the finite-difference scheme to accommodate the large variations of concentrations are discussed. An effort has been made to avoid any subjective analysis scheme for the preparation of the input data.

The model was used to study SO_2 distributions in the St. Louis metropolitan area during 25 consecutive days in February 1965. The computed results were evaluated with respect to observed data by using various statistical methods. The computed results agree favorably with experimental measurements for both long-term and short-term average concentrations. Computations also indicate the model's capabilities and flexibilities for dealing with the rapid variations of atmospheric conditions. The advantages and limitations of the model are also discussed.

1. Introduction

In recent years, a number of urban air quality diffusion models have been developed. Such models can be used to study the complicated relationships between air quality and emission sources as a function of various parameters, *viz.* meteorological and surface conditions. The aim of these models is toward air quality prediction as well as long-term air quality management planning. In view of the long standing air pollution problems and growing emphasis on air quality improvements, the subject has great practical importance. However, the present state of the art in diffusion modelling raises some doubts about the possibility and feasibility for achieving these goals. The objective of this work is to develop a model through which we can study thoroughly the relation between air quality and sources as well as the feasibility of its eventual use in air quality prediction and management. In general, these models are formulated by the use of a concentration equation governing the pollutant mass which are based on the physical principle of mass conservation. The governing concentration equation, supposedly, can be solved for given input of the source emission rates, meteorological and surface conditions, turbulent transport mechanism, and transformation rates. However, due to the complex process of the air motion in an urban atmosphere and inadequate data acquisition, the required detailed information for the input is not available. Thus, for practical purposes, various approaches have been developed, based on either statistical theory, or phenomena, or even arbitrary assumptions. The choice of the approach, more often than not, depends on its applicability and convenience.

The simplest formulation is the box-type diffusion model (Frenkiel, 1956; Leahey, 1972) which assumes the vertical pollutant distribution to be uniform inside the atmospheric mixing layer. It offers a quick result for long-term concentrations.

Another type of simple model reported by Gifford and Hanna (1970) is based on the similarity hypothesis. This approach assumes that the vertical concentration distribution is similar with respect to a height scale along the wind axis. This height, called the height of the polluted air, is determined by the requirement of separation of variables in the governing equation. The application of the model gave some good results for long-term averaged concentrations. A somewhat similar approach with more complicated formulation was reported by MacCracken *et al.* (1972).

The best known of the practical equations based on statistical theory is the Gaussian plume formula which was evolved from the solution of the concentration equation under isotropic, homogeneous condition. It assumes that at downwind from a point source, the ensemble average concentration on the crosswind plane will approximate a normal statistical distribution. At present, many existing urban diffusion models are primarily based on Gaussian formulations. The models developed by Turner (1964), Roberts *et al.* (1970), Johnson *et al.* (1970), Shieh *et al.* (1972) are a few typical examples. Reviews of existing models are reported by Moses (1969) and Neiberger (1971).

On the other hand, the concentration diffusion equation has not been widely used in this kind of study. When properly applied, the equation can give a better description of atmospheric diffusion processes. There are various solutions of this equation for point sources with the specified atmospheric conditions, such as those of Rounds (1955), Iordanov (1966), Shir (1970, 1972a) and many others. Its application to a multiple-source urban diffusion model is still considered to be at a beginning stage. A few models, such as those of Harrington (1965), Sklarew *et al.* (1972), Lamb and Neiburger (1971), Randerson (1970) and Reynolds *et al.* (1973) have been developed, but a more extensive test of this approach had yet to be made.

It is well known that the major factors that characterize diffusion processes in the atmosphere are the state of atmospheric turbulence and its underlying surface properties. It is apparent that the Gaussian plume formula is not flexible enough to include all possible variations that the air motion experiences under urban atmospheric conditions. If a model has to accommodate the temporal and spatial variations of meteorological parameters, effects of the inhomogeneous surface conditions and other features, a realistic approach as we see it, lies in the application of the concentration diffusion equation. It is possible that the availability and accuracy of input data for the models, both for meteorological parameters and source inventory, will be improved in the future. Therefore, application of this kind of model would be far more advantageous.

In this study, we propose a new urban diffusion model, based on a concentration diffusion equation. The major purposes of this study are the following:

1. To investigate the feasibility of using concentration diffusion equation for a diffusion computation which involves multiple source emissions.
2. To develop a numerical technique which is to handle the inhomogeneity of multiple-source emission rates, the complex urban atmosphere, and the change of surface boundary conditions.
3. To obtain a practical method such that the model can utilize current available meteorological and source inventory data.
4. To develop a general method such that the model computation will not require any subjective analyses, arbitrary adjusted parameters, and "tune-up."

2. Input data

The basic data used in this study were obtained from the Division of Meteorology, Air Program Office, U. S. Environmental Protection Agency. Originally, the data were collected for the purpose of St. Louis SO_2 dispersion model study as part of the Interstate Study (Venezia and Ozolin, 1966), covering the period from 1 December 1964 to 28 February 1965. The data consists of source emission inventory, meteorological variables, and SO_2 concentrations at monitoring stations. It should be noted that the original concept of the St. Louis SO_2 dispersion study was for the developing of a Gaussian-type model. Consequently, this set of data was gathered for this specific purpose. The detailed description of these data can be found in Turner and Edmisten (1968). However, these data are not compatible with the requirement of the gradient transport type of diffusion model, and additional analysis was therefore necessary. The following discussion will briefly describe the basic data.

a. SO_2 *source emission data*

The St. Louis, Missouri–East St. Louis, Illinois Metropolitan area is divided into 1200 square area source grids, 30 squares in the east–west direction and 40 squares in the north–south direction (see Fig. 1). The dimensions of the grids are 5000 ft on the sides. In addition, Fig. 1 also shows 44 industrial point sources within the region. No adjustments have been made to the emission data when applied in our diffusion computations.

The time-dependent source emission rate (gm sec^{-1}), averaged over a 2-hr period, for each area source and point source are computed by equations which can be found in Turner and Edmisten (1968). The method of calculating these emission rates and its validation have been discussed by Turner (1968). No adjustments have been made to the emission data when applied in these diffusion computations.

In addition to source emission data, two additional sets of data are included. First, the average building height of each area source grid was used to estimate the surface roughness. Second, the stack height of point sources and the values of plume rise time wind speed (m^2 sec^{-1}) are also included. For cases when no plume rise information was given, the plume rise was estimated from the SO_2 emission.

b. Meteorological data

Hourly averaged meteorological data are routinely obtained at two airports in St. Louis area: Lambert

Field and Scott Field. The data consist of surface wind direction and speed, sky cover, surface temperature, visibility and weather conditions, etc. The hourly averaged surface temperature, wind direction and speed are also measured at three stations on the periphery of the urban area. They are located at Lindbergh High School, the Missouri State police station, and Hazelwood High School. The instruments' specifications and operation were described by McElroy and Pooler (1968). In addition, the instruments which measured wind speed, direction and temperature were installed at KMOX-TV tower located in downtown St. Louis (see McElroy and Pooler, 1968). This operation consists of three levels of instrumentation and one level of bivane standard deviation. The location of measurement sensor stations is given in Fig. 2.

c. Mixing heights

The data of two daily mixing heights, namely, one morning minimum and one afternoon maximum height, estimated by EPA were used. These heights represent the level at which the adiabatic lapse rate (based on estimated surface urban temperature) intersects the rural temperature profile measured by morning radiosondes. This intersection height may be in error due to an ill-defined temperature profile or over-estimated urban surface temperatures. The average range of heights is 300 m to about 600 m.

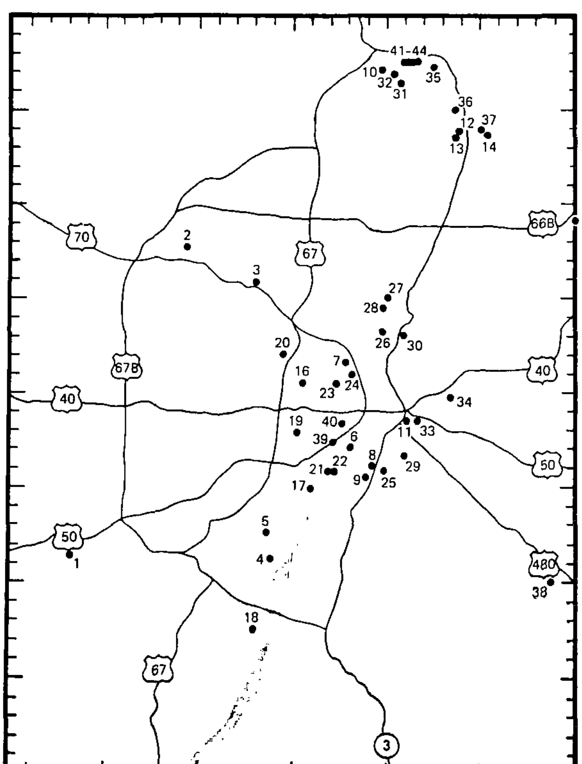


FIG. 1. Geographical distribution of area and point sources for the St. Louis metropolitan area.

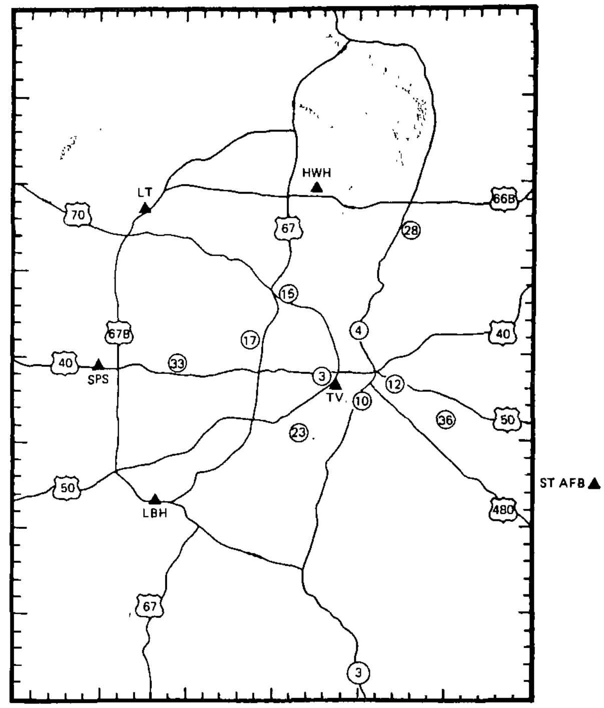


FIG. 2. Locations of meteorological stations and the SO_2 monitoring network in the St. Louis metropolitan area.

d. SO_2 *concentration*

Fig. 2 also shows the sampling stations where SO_2 concentrations are measured. Original data consisted of 40 stations where concentrations were obtained for a 24-hr averaging period. At ten of these stations additional 2-hr averaged concentrations were made. The detailed description of the instrumentation and its operation are given by Farmer and Williams (1966). The validation of this model is based on the latter 10 stations where 2-hr averaged concentrations were measured.

3. Model formulation

a. Equations, boundary conditions and initial conditions

The region of interest is the St. Louis metropolitan area. This area encompasses 30×40 square area source grids based on the emission inventory made by Turner and Edmisten (1968) (Fig. 1). The time varying mixing depth is taken as the upper limit of the model. Since this area has a reasonably flat terrain, the effects of topography are neglected. The surface roughness parameter is used to represent the effects of urban buildings. SO_2 emission is considered passive and therefore does not alter the meteorological conditions. The turbulent diffusion of SO_2 is assumed to be of the gradient diffusion type. The governing equation of SO_2

in the atmosphere based on the conservation law is

$$\frac{\partial C}{\partial t}+\nabla\cdot \mathbf{V}C=K_H\nabla_H{}^2C+\frac{\partial}{\partial z}K\frac{\partial C}{\partial z}+Q+R, \tag{1}$$

where C is the mean concentration of SO_2, $\mathbf{V}=(U,V,W)$ is the mean wind vector, Q the source strength rate, R the chemical reaction rate, K_H the horizontal eddy diffusivity, $\nabla_H{}^2$ the horizontal Laplacian operator, and K the vertical eddy diffusivity. The boundary conditions are:

$$K\frac{\partial C}{\partial z}=0, \quad z=0, H \tag{2}$$

$$\frac{\partial^2 C}{\partial x^2}=0, \quad x=0, x_{\max} \tag{3}$$

$$\frac{\partial^2 C}{\partial y^2}=0, \quad y=0, y_{\max}. \tag{4}$$

Here $H=H(t)$ is the mixing depth of the planetary boundary layer, and $x_{\max}$ and $y_{\max}$ are the east and north boundaries of the area respectively. The boundary surfaces above and below are assumed impermeable to the SO_2. The absorption of the SO_2 by the ground surfaces is neglected. The exact lateral boundary conditions are not available. Those conditions in (3) and (4) which extrapolate the concentration outside the region serve as a reasonable approximation when the region of computation is large enough. In practice, and as we have found, the lack of well-posed boundary conditions does not cause serious problems. This is because the horizontal advection terms, which dominate the horizontal diffusion terms, are only first order in the space derivative. The computation may be affected by the inflow boundary conditions. However, if there is no high concentration outside the inflow boundary this influence is minimum, and the linear extrapolation could offer a fair approximation. In this study, the region computed is about ten times that of the urban area where the major sources are located. Moreover, there is neither any nearby major urban area nor any return flow to influence the inflow boundary conditions. The initial conditions are arbitrarily set to zero. The computations show that the concentrations reach observed levels within approximately 2 hr under average wind speed conditions. Hence the initial conditions are not important for the concentration computations after this initial time. This required time period depends on the wind speed and the region of interest. This time interval can be estimated by $T_u\approx L/\bar{u}$, where L is the distance downwind from the sources and $\bar{u}$ the average wind speed. However, this may not be valid when the wind changes direction drastically during this time period. The model simulation begins at 1400 Local Standard Time (LST) which corresponds to the time period used in the source emission inventory and data acquisition system.

b. Physical parameters

The parameters required for the integration of Eq. (1) are $\mathbf{V}$, K_H, K, Q, R and H which, except for Q, are not provided explicitly or sufficiently. The following discussions outline the methods which were used to obtain the input data for the model requirement.

1) Wind field

The wind vector $\mathbf{V}=\mathbf{V}(x,y,z,t)$ is required at every grid point for each time step of integration. The hourly averaged surface wind field for the total region was obtained by using a weighted interpolation scheme. Data from the measurement stations were interpolated to a square grid, which had a size of five area source dimensions. Several schemes have been tested in this analysis. It was found that reasonable results can be obtained from the equations (Wendell, 1970):

$$\bar{u}_{ij}=\sum_{m,n}(\tilde{u}_{mn}/r_{mn}^2)/\sum_{m,n}(1/r_{mn}^2), \tag{5a}$$

$$\bar{v}_{ij}=\sum_{m,n}(\tilde{v}_{mn}/r_{mn}^2)/\sum_{m,n}(1/r_{mn}^2), \tag{5b}$$

where $\bar{u}_{ij}$ and $\bar{v}_{ij}$ are components of the wind vector at analysis grids in the x and y directions, respectively; and $\tilde{u}_{mn}$ and $\tilde{v}_{mn}$ are the initial guess fields at analysis grid and r_{mn} the distance from grid (i,j) to grid (m,n). The initial guess field is obtained by assuming

$$\left.\begin{aligned}\tilde{u}_{mn}&=u_k\\ \tilde{v}_{mn}&=v_k\end{aligned}\right\}\text{for minimum of } r_k,$$

where u_k and v_k are components of the wind vector measured at station k and r_k is the distance from a grid to station k. From this analyzed wind field a linear interpolation is employed to obtain a wind vector at each grid point to be used in the numerical scheme.

The mesoscale wind field analysis is a very difficult problem. In reality, we don't think this simple method will be able to handle all possible meteorological conditions. Due to the limited number of measurement stations, a more complex method is not applicable in our study. In this study, an attempt was made to avoid any subjective analysis methods. Thus, throughout our computations the same analysis procedure was applied without any subjective adjustments. In the later part of our study we became aware of the existence of a complete subjectively analyzed wind field for this period (Turner, 1972, personal communication). The model performance based on this wind field as compared to our method will be discussed in Section 5.

The spatial distribution of upper layer wind data was not available, and the present knowledge of urban meteorology cannot offer much information about it.

Fortunately, the concentrations at those locations where the local sources dominated are not sensitive to the upper wind. The vertical wind profiles at each grid location are assumed to be of power law form

$$|\mathbf{V}| = |\mathbf{V}_s| (z/z_s)^p, \quad (6)$$

where $\mathbf{V}$ and $\mathbf{V}_s$ are the upper and surface wind at the height z and z_s respectively. The power constant p is determined by

$$p = \ln(|\mathbf{V}_3|/|\mathbf{V}_1|)/\ln(z_3/z_1), \quad (7)$$

where $\mathbf{V}_3$ and $\mathbf{V}_1$ are the winds at the height z_3 (140 m) and z_1 (39 m) of the TV tower. The values of p are restricted between 0.15 and 0.65 which are the usually observed values. The directions of the upper wind are also unknown. It is understood that the upper wind has a direction to the right of the surface wind in the Northern Hemisphere. However, this angle cannot be determined quantitatively from any theory under general conditions. Two methods have been tested. One method assumes no directional change and the other assumes equal angle change with height over each height interval from z_s to z_3 with the same total angle change as that measured at the TV tower. Results from both methods differ very little. Hence the assumption of constant wind direction with height was used for the computations. The vertical winds were calculated from horizontal winds through the continuity equation. Since the interpolated horizontal winds are so smooth, the vertical winds do not significantly influence the concentration distributions. This is in contrast to the case when a strong urban heat island occurs. However, the incomplete knowledge of the vertical winds resulting from the urban heat island effects make this assumption necessary.

2) ATMOSPHERIC STABILITY

It is well known that turbulent diffusion of air pollutants is directly related to the atmospheric turbulent intensity which is categorized by the stability of the atmosphere. But what is the practical way to describe the stability? From planetary boundary layer theory, the Richardson number Ri or Monin-Obukhov length L is used to represent the stability.

The availability of temperature and wind observation on the TV tower made it possible to compute a bulk Richardson number. McElroy and Pooler (1968) suggested that atmospheric stability can be categorized by a bulk Richardson number and bivane standard deviation. Our first attempt was to compute eddy exchange coefficients as a function of bulk Richardson number, bivane standard deviation, and surface roughness. It was found that this method gave inconsistent results. The deficiencies of this method are three-fold. First, the computed value of the Richardson number depends on the accuracy of the vertical temperature gradient measured at TV tower. Second, the TV tower is located in downtown St. Louis, and the parameters measured at this location may be influenced by nearby high buildings and local heat emission. Thus, the stability parameters based on these measurements may only represent the local stability condition. Third, the correlations between bulk Richardson number and bivane standard deviation are very poor. In the following discussion, we will present a method which is used in the final computation. This approach will enable an urban diffusion model to be based on the concentration diffusion equation without the benefit of vertical temperature measurements.

The basic stability classification proposed by Pasquill (1962) is employed. The wind speed at the lowest level of TV tower measurement (39 m) and sky cover observed at the two airports were used to establish the stability class. Pasquill's stability category is a discrete function (represented by stability classes A–F). It will be replaced by stability classes −3, −2, −1, 0, 1 and 2 in our discussion. (The stability class 0 represents neutral conditions.) Moreover, the stability class was evaluated as a continuous function by interpolating Pasquill's class with respect to wind speed. However, in the transient period, after sunrise and sunset, the past history of stability must also be considered. The summary of this interpolation scheme is given in Table 1.

TABLE 1. Key used to estimate the continuous stability classification.*

Mean wind speed (m sec^{-1})	Day: Incoming solar radiation, Strong	Day: Incoming solar radiation, Moderate	Day: Incoming solar radiation, Slight	Transient period: Day→	Transient period: ←Night	Night: Thinly overcast or 4/8 low cloud	Night: Cloud $\leqslant$3/8
<2	−3.5∼−3.0	−3.0∼−2.2	−2.5∼−2.0	−1.5∼−0.5	0.5∼1.5	1.5	2.5
2∼3	−3.0∼−2.2	−2.2∼−2.0	−2.0∼−1.0	−1.0∼−0.3	0.5∼1.0	1.5∼0.6	2.5∼1.6
3∼5	−2.2∼−1.5	−2.0∼−1.0	−1.0∼−0.5	−1.0∼−0.3	0.3∼0.5	0.6∼0.3	1.6∼0.5
5∼6	−1.5∼−1.0	−1.0∼−0.3	−0.5∼−0.2	−0.4∼−0.2	0.2∼0.4	0.3∼0.1	0.5∼0.3
6∼8	−1.0∼−0.3	−0.3∼−0.1	−0.2∼ 0	−0.3∼−0.1	0.1∼0.2	0.1∼0	0.3∼0.1
>8	−0.3	−0.1	0	0		0	0.1

* Note: 1) See Turner (1969) for overall explanations.
2) Zero index is equivalent to neutral condition.
3) An intermediate value is computed by a linear interpolation scheme according to the specified range of wind speed.

Recently, the relation between the specified Pasquill-Gifford stability classes and the physical parameters of turbulence was studied by Golder (1972). Golder found the relations between the Pasquill-Gifford's stability classes and the Monin-Obukhov length with surface roughness as an additional parameter. Golder's data can be approximated by the expressions

$$1/L=\pm[d\ln(1.2+10/z_0)]^2 10^{f(S)}, \tag{8}$$

$$f(S)=-a/(1+b|S|^c), \tag{9}$$

where S is the stability class; $a=4$, $b=1.3$, $c=0.85$, $d=0.216586$; and $z_0=z_0(x,y)$ is the surface roughness parameter in meters. The stability parameter $S=0$ denotes neutral conditions, and negative and positive values of S denote unstable and stable conditions, respectively. Thus, the sign of L in (8) must be the same as that of S. Moreover, the continuous values of S as discussed previously allow smooth changes between stability classes. Since this method is much easier to use than measuring vertical temperature distributions, its merits should not be overlooked. Certainly, it may require further improvements such as continuous insolation classifications, non-uniform spatial stability distributions, and effects of precipitations.

3) Eddy diffusivity

To calculate eddy diffusivities of air pollutants is a classical problem. Not much information about their behavior are known except near the surface. Recently, Shir's turbulence transport model (1973) successfully calculated the turbulence structure in the planetary boundary layer. The vertical distribution of K under neutral conditions can be expressed as

$$K=u_*l,\quad l=k_0ze^{-4z/H}, \tag{10}$$

where u_*, k_0 and H are friction velocity, the von Kármán constant, and the height of the boundary layer, respectively. We assume that $K=K_sl/l_s$ for non-neutral conditions, where the subscript s denotes the values at $z=10$ m. From given L and $\mathbf{V}_s$ (wind velocity at $z=10$ m), we now calculate K_s from the following two procedures:

1. Calculate u_* from $\mathbf{V}_s$, L and z_0 by

$$u_*=k|\mathbf{V}_s|/\psi_m, \tag{11}$$

where $\psi_m=\int_{z_0}^{z}(\phi_m/\bar{z})d\bar{z}$, ϕ_m being the non-dimensional wind shear. According to Businger *et al.* (1971), ϕ_m can be expressed as

$$\phi_m=\begin{cases}1+\alpha\zeta, & \zeta>0\\ (1-\beta\zeta)^{-\frac{1}{4}}, & \zeta<0\end{cases} \tag{12}$$

where $\zeta=z/L$, $\alpha=4.7$ and $\beta=15$. Using Eq. (12), we have

$$\psi_m=\begin{cases}\ln(1+z/z_0)+\alpha\zeta, \quad \text{for}\quad \zeta>0\\ \ln(1+z/z_0)-2\ln[1+\omega/2]-\ln[(1+\omega^2)/2]\\ \qquad +2\tan^{-1}\omega-\pi/2, \quad \text{for}\quad \zeta<0\end{cases} \tag{13}$$

where $\omega=1/\phi_m$.

2. Calculate K from u_* and ϕ_n by

$$K=u_*l/\phi_n, \tag{14}$$

where

$$\phi_n=\begin{cases}\gamma+\alpha\zeta, & \zeta>0\\ \gamma(1-\beta'\zeta)^{-\frac{1}{2}}, & \zeta<0\end{cases}$$

and $\gamma=0.74$, $\beta'=9$. The formulas (12)–(14) are valid in the surface layer. We calculate $K_s=K$ ($z=10$ m) and then extrapolate K to higher altitude by $K=K_sl/l_s$. This approach by no means implies that the turbulent diffusion of air pollutants over an urban area is really so simple. Those formulas are derived from experimental data based on equilibrium turbulence which may not occur over an urban area where the horizontal inhomogeneity plays an important role. Shir (1972b) found that turbulence is non-equilibrium in the vicinity of a change in surface roughness. However, little information is available about the urban effects on the eddy diffusivity. The formulas used here are subject to improvement when more knowledge about them is available. However, we feel that it is a significant improvement over existing methods because the effects of surface roughness are taken into account in the calculation of L and K. Shir (1972a) pointed out that the effects of surface roughness can influence the concentration distribution significantly. These effects are neglected in the Gaussian plume approach.

The horizontal eddy diffusivity K_H has no significant effect on the results and its behavior is not well understood, especially under stable conditions. In the present study, a value of 500 m^2 sec^{-1} was assumed.

4) Surface roughness

Although surface roughness affects the turbulent dispersion and wind profiles, no surface roughness measurements were made over the St. Louis urban area. Lettau (1970) proposed a simple formula to estimate the roughness length of the urban area by

$$z_0=0.5rh, \tag{15}$$

where r is the silhouette area ratio and h the effective height of the roughness elements. The values of r range from about 0.05 to 0.5 from rural to urban areas. The values of r are assumed to be proportional to the area emission sources density and are estimated as

$$r=0.04[1.0+0.37Q_a(x,y)/\bar{Q}], \tag{16}$$

where Q_a is the mean area source strength and $\bar{Q}$ the space average area source strength. This gives values of r ranging from 0.04 to 0.5 and values of z_0 from 0.4 m in suburban areas to 6 m in urban areas. These values offer a crude estimate of the roughness distributions in the St. Louis area. This formula is subject to improvement when more information is available on the effect of different urban areas on the values of r and z_0.

5) THE MIXING HEIGHT

Turbulent mixing plays an important role in the dispersion of pollutants in the atmospheric layer under the mixing height. This height may not always be the height of inversion layer. For instance, in a neutral atmosphere, there is a mixing height but no inversion height. For a horizontally homogeneous planetary boundary layer under neutral conditions, the mixing height is about equal to half the height $h_* \equiv u_*/f$, where f is the Coriolis force parameter (Shir, 1973). However, the technique used to estimate the mixing height is actually that of estimating the inversion height.

The hourly mixing height was estimated by interpolation from the given minimum and maximum heights. The period of minimum height is assumed to last from the 0000 to 0600 LST and the maximum height from 1400 to 1800 LST. The morning minimum height was restricted to be lower than the previous day's maximum. The maximum height was kept less than 1200 m, because any height above that has little influence upon the surface concentrations within the region of investigation under normal wind speed. This can be shown by comparing the advection time scale T_u to the diffusion time scale T_d. Here $T_u = L/\bar{u}$, where L is the length of the urban area and $\bar{u}$ the average surface wind speed at $z = 10$ m; and $T_d = H^2/K$, where H is the mixing height and K the average eddy diffusivity. Under neutral conditions, $K = u_* l = k_0\bar{u}/(\ln 10) \approx 6\bar{u}$; with $l \approx 30$ m this gives $\beta_t \equiv T_u/T_d \approx 6L/H^2$. In this study, $L \approx 20$ km, thus, $\beta_t \approx 0.1$ for $H = 1000$ m and $\beta_t \approx 1$ for $H = 300$ m. Therefore, when the mixing height is greater than 1000 m, the time of advection is much shorter than the time of diffusion, and the effects of the reflection from the inversion base become small.

6) CHEMICAL REACTION RATE

The chemical reaction rates of SO_2 can be expressed by $R = -k_a C$, where k_a is the reaction rate constant. The reaction rate depends on the insolation and the atmospheric concentrations of water vapor, nitric oxide, hydrocarbon and particulates. Its estimated values range from 10^{-3} to 10^{-7} sec^{-1} (Leighton, 1961). Wilson and Levy (1968) found values of k_a ranging from 0.8×10^{-4} to 1.4×10^{-4} sec^{-1} in smog chamber experiments with medium relative humidity. The value of $k_a = 10^{-4}$ sec^{-1} was used in this study. For normal wind speeds (5 m sec^{-1}), it is believed that the reaction rate is slower than the advection rate of wind over the medium size urban area. This can be estimated by $\gamma_t \equiv T_u/T_c$, where $T_c = k_a^{-1}$ is the time constant of reaction rate and $T_u = L/\bar{u}$ is the time scale of advection. The chemical reaction effects will be pronounced when $\bar{u} \leqslant L/T_c$. In St. Louis, $L \approx 20$ km covers most of measurement stations. If $T_c \approx 3$ hr, then $\bar{u} \approx 2$ m sec^{-1}. Thus, the chemical reaction effects are significant under low wind speed conditions ($\bar{u} \leqslant 2$ m sec^{-1}).

4. Methods of analysis

a. Grid specifications

Our three-dimensional grid system consists of $30\times40\times14 = 16800$ grid points. The x, y, z axes are oriented east-west, north-south, and vertically, respectively. The horizontal grid sizes $\Delta x = \Delta y = 5000$ ft $= 1524$ m were chosen according to the grid size of the emission source inventory. The vertical grid sizes are specified as follows:

$$\Delta z_k = \begin{cases} 20 \text{ m}, & 1 \leqslant k \leqslant 5 \\ 25 \text{ m}, & 6 \leqslant k \leqslant 9 \\ (H-200)/4, & 10 \leqslant k \leqslant 13, \quad \text{for } H \geqslant 300 \text{ m} \end{cases}$$

where H is the mixing height when $H \geqslant 300$ m. The lower nine grid points under 200 m have a fixed spatial size because the effective heights of point and area sources are within this layer. The levels of the upper four grid points are determined by the hourly varying mixing height. The grid spacing for these four grid points is set to be larger than or equal to 25 m. Thus, the minimum height of the grid system is 300 m. When the mixing height is lower than the top of the grid system, small values of eddy diffusivity were forced at those grids located above the mixing height.

b. Numerical methods

A second-order, central finite-difference scheme was used to integrate the advection and horizontal terms and the Crank-Nicholson method was used for the vertical diffusion term. Since the concentration fields usually have large variations, phase errors resulting from the finite-difference method are large. Therefore, careful treatment is required. The finite-difference approximation to the concentration equation is

$$\begin{aligned} C_{ijk}^{n+1} = {} & C_{ijk}^{n} + \mathfrak{U}[C_{ijk}^{n}] + \mathfrak{V}[C_{ijk}^{*}] + \mathfrak{W}[C_{ijk}^{**}] \\ & + D_{xy}[C_{ijk}^{n}] + D_z[C_{ijk}] + \Delta t Q_{ijk}/\Delta V - \Delta t k_a C_{ijk}^{n}, \end{aligned} \tag{17}$$

where $\Delta V = \Delta x \Delta y \Delta z$ is the grid volume; $\mathfrak{U}$, $\mathfrak{V}$, $\mathfrak{W}$ are finite-difference operators for horizontal and vertical advection, and D_{xy}, D_z for horizontal and vertical diffusion; n is the time step index; and i, j, k are grid indexes along the x, y, z directions, respectively.

The operators $\mathcal{U}$, $\mathcal{V}$ and $\mathcal{W}$ are defined by

$$\left.\begin{aligned}\mathcal{U}[C_{ijk}]&=F_x[C_{ijk}]-F_x^*[C_{ijk}]\\ \mathcal{V}[C_{ijk}]&=F_y[C_{ijk}]-F_y^*[C_{ijk}]\\ \mathcal{W}[C_{ijk}]&=F_z[C_{ijk}]-F_z^*[C_{ijk}]\end{aligned}\right\},$$

where

$$F_x[C_{ijk}]=\frac{\alpha_{xi}}{(1+r_{xi})}[(1+\alpha_{xi})C_{i-1,jk}+(1-\alpha_{xi})C_{ijk}], \quad (18a)$$

$$F_x^*[C_{ijk}]=\frac{\alpha_{xi}^*r_{xi}}{(1+r_{xi})}[(1+\alpha_{xi}^*)C_{ijk}+(1-\alpha_{xi}^*)C_{i+1,jk}], \quad (18b)$$

$$r_{xi}=(\Delta x_i/\Delta x_{i-1})\alpha_{xi}=\Delta tU_{i-\frac{1}{2},jk}/\Delta x_{i-1}, \quad (18c)$$

$$\alpha_{xi}^*=\Delta tU_{i+\frac{1}{2},jk}/\Delta x_i.$$

The similar expressions for F_y and F_y^* can be obtained by interchanging the index i by j in (18). Similarly, for F_z and F_z^*, we replace x and changes of i by z and k respectively. Eqs. (18) apply to non-uniform grid spacings. Since grid space is uniform in the x and y directions, $\Delta x_i=\Delta x$, $\Delta y_j=\Delta y$, $r_{xi}=1$ and $r_{yj}=1$. The time-splitting method was employed in which the new values of C calculated from $\mathcal{U}$ operator are used for $\mathcal{V}$ operator. Thus

$$\left.\begin{aligned}\overset{*}{C}_{ijk}&=\overset{n}{C}_{ijk}+\mathcal{U}[\overset{n}{C}_{ijk}]\\ \overset{**}{C}_{ijk}&=\overset{*}{C}_{ijk}+\mathcal{V}[\overset{*}{C}_{ijk}]\end{aligned}\right\}.$$

Since the distributions of C are not smooth, a special treatment is required to prevent the truncation error from amplifying. The following simple method was used:

$$F_x=\begin{cases}C_{ijk}, & \text{if} \quad \alpha_{xi}<0, \quad C_{ijk}<-F_x\\ C_{i-1,jk}, & \text{if} \quad \alpha_{xi}>0, \quad C_{i-1,jk}<F_x\end{cases} \quad (19a)$$

$$F_x^*=\begin{cases}C_{ijk}, & \text{if} \quad \alpha_{xi}^*>0, \quad C_{ijk}<F_x^*\\ C_{i+1,jk}, & \text{if} \quad \alpha_{xi}^*<0, \quad C_{i-1,jk}<+F_x^*\end{cases} \quad (19b)$$

Similar treatment is also applied to F_y, F_y^*, F_z and F_z^*. This method can preserve the amount of concentration when it is transported from one grid to the next. In addition, it can prevent the values of concentration from becoming negative due to truncation errors. The operators D_{xy} and D_z are defined by

$$D_{xy}[C_{ijk}]=\Delta tK_H[C_{i-1,jk}-2C_{ijk}+C_{i+1,jk})/\Delta x^2 + (C_{i,j-1,k}-2C_{ijk}+C_{i,j+1,k})/\Delta y^2], \quad (20)$$

$$D_z[C_{ijk}^{n+1}]=\gamma_k(\theta D[C_{ijk}^{n+1}]+(1-\theta)D[C_{ijk}^n]), \quad (21)$$

where

$$D[C_{ijk}]=r_kK_{k-\frac{1}{2}}[C_{ij,k-1}-(r_kK_{k-\frac{1}{2}}+K_{k+\frac{1}{2}})C_{ijk}+K_{k+\frac{1}{2}}C_{ij,k+1}],$$

$$\gamma_k=\Delta t/[\Delta z_k(\Delta z_k+\Delta z_{k-1})],$$

$$r_k=\Delta z_k/\Delta z_{k-1},$$

and θ is a parametric constant.

The stability criteria for the terms of advection, horizontal and chemical reaction rate are well known (Ritchmyer and Morton, 1967) as

$$\alpha_a,\ \alpha_a^*<1, \text{ where subscript } a \text{ denotes } x, y, \text{ or } z,$$
$$\Delta tK_H/(\Delta x^2+\Delta y^2)\leqslant\tfrac{1}{4}, \text{ and } \Delta tk_a\leqslant 1.$$

The stability criterion for the vertical diffusion term for uniform Δz and K is

$$K\gamma_k\begin{cases}\leqslant\dfrac{1}{4(1-2\theta)}, & \text{for} \quad \theta\leqslant 1/2\\ >0, & \text{for} \quad \theta>1/2\end{cases} \quad (22)$$

This stability condition simply says that the implicit scheme used in (21) is unconditionally stable when $\theta\geqslant\frac{1}{2}$. However, it cannot prevent amplification of truncation errors when the concentration fields are not smooth and the value of γ_k is large. The present study indicates that another additional condition is required. When the diffusion takes place between two grid boxes, the flux across the boundary is proportional to $-K\partial C/\partial z$. The flux will cease when the gradient approaches zero. Thus, we have

$$(C_{ijk}^{n+1}-C_{ij,k+1}^{n+1})/(C_{ijk}^n-C_{ij,k+1}^n)\geqslant 0.$$

This requirement leads to the condition

$$K_{k+\frac{1}{2}}\gamma_k\leqslant\frac{1}{4(1-\theta)}. \quad (23)$$

Hence, the value of θ must be close to 1, if a large value of γ_k is used. Eq. (17) applies to a constant grid spacing system. When the grid space is a function of time, a correction due to the changing volume is needed. It can be shown that $-C_{ijk}\partial \ln\Delta z_k/\partial t$ should be added on the right-hand side of Eq. (17). The implementation of this correction is simplified by changing the height $H(t)$ for each hourly period. With this arrangement, the concentrations at the upper five grid points are adjusted hourly according to the height H as follows:

$$\overset{n+1}{C}_{ijk}=\begin{cases}\overset{n}{C}_{ijk}/s_n, & k\geqslant 11\\ \left(\dfrac{1+r_k/s_n}{1+r_k}\right)\overset{n}{C}_{ijk}, & k=10\end{cases}$$

where $s_n=(H^{n+1}-z_{10})/(H^n-z_{10})$.

5. Results and discussions

The period from 1–26 February 1965 was chosen for these model computations. During these 25 consecutive days, various meteorological conditions occurred which allow evaluation of the generality and performance of the model. The computer time for this computation is 2–5 min for every 24-hr simulation on IBM system 360/195.

a. General assessment

The period-average concentration for 25 days at ten monitoring stations (indicated by station number) are shown in Fig. 3. Both measured values from 2-hr and 24-hr instruments are used for comparison. The 2-hr data are consistently larger than the 24-hr data. The agreement between computed and measured values is very good. The correlation coefficient is 0.899 and 0.873 for 2- and 24-hr data, respectively. Excellent agreement can be seen for stations 15, 28, 33, 36, fair for stations 3, 4, 12, 17, and poor for station 10. The three-month average results from the Gaussian plume model (Pelton *et al.*, 1972) are also shown. The present results for shorter averaging period (25 days) are better than those from the Gaussian model which yields the correlation coefficient of 0.675 based on the regression line. Interestingly enough, the stations where two models either overestimate (stations 3, 12, 23) or underestimate (stations 4, 15, 17, 10) are consistent. The underestimation at station 10 may be due to the influence of nearby large stacks which contributed to local high concentration values. Such strong local variations cannot be resolved with the present grid model. The stations (4, 15, 17) where the model underestimated concentration values are located on the northwestern part of the urban area. On the other hand, concentration at those stations located on the southeastern side of the city were overestimated by the model computations. This geographical dependence of model performances (including Gaussian plume model) indicates that further study is warranted.

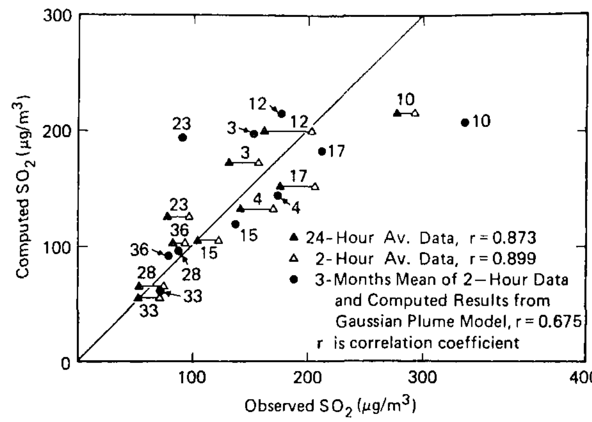


FIG. 3. Comparison of observed and computed 25-day averaged SO_2 concentration values (1–26 February 1965).

These phenomena may be due to the urban heat island effect which was not properly accounted for in the model. Since the prevailing wind during this period was from the northwest, the underestimated concen-

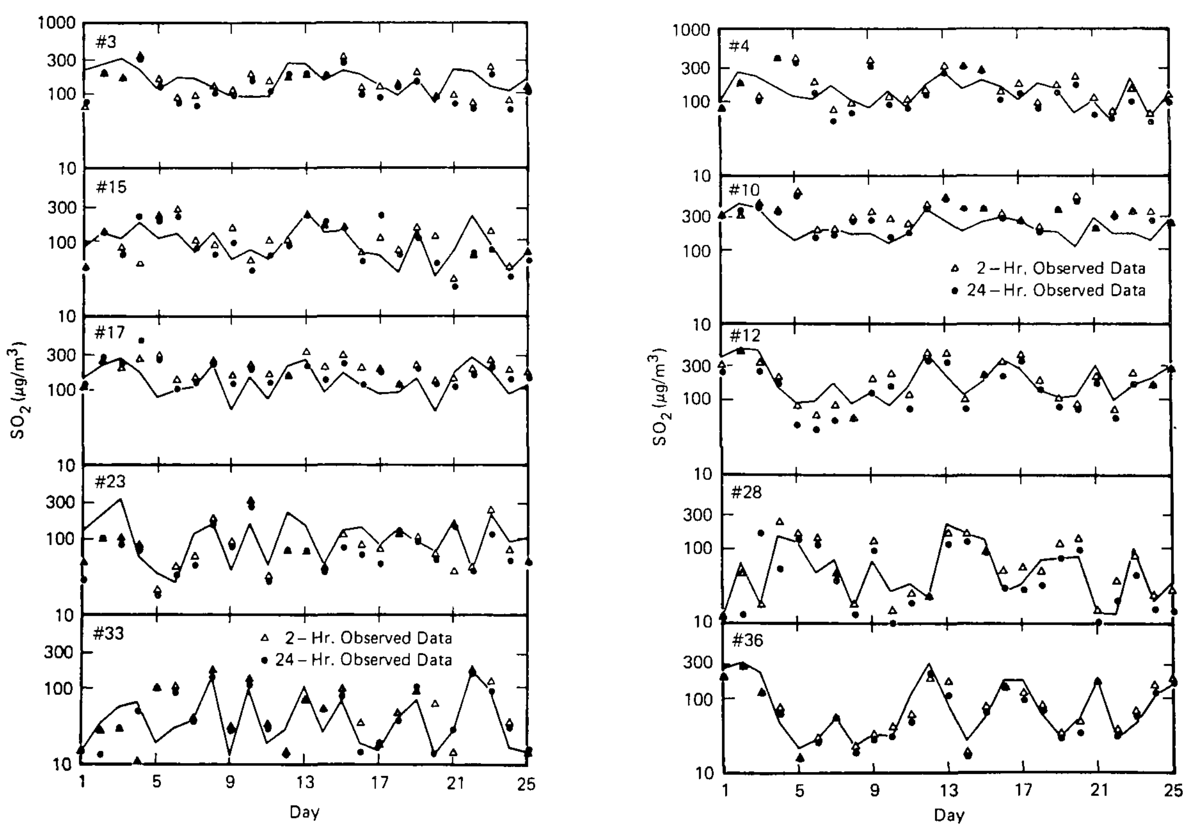


FIG. 4. Comparison of observed and computed 24-hr averaged variations of SO_2 concentrations for the 25-day period at each monitoring station.

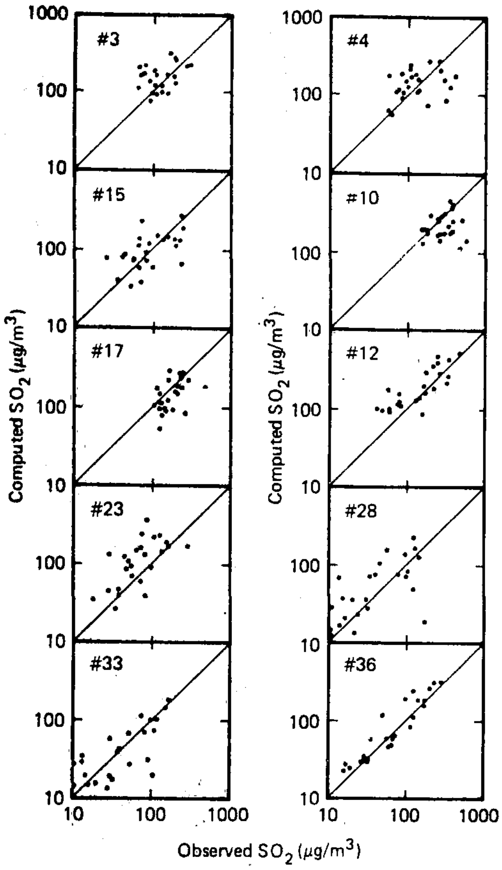


FIG. 5. Comparison of observed and computed 24-hr averaged SO_2 concentration values at each monitoring station.

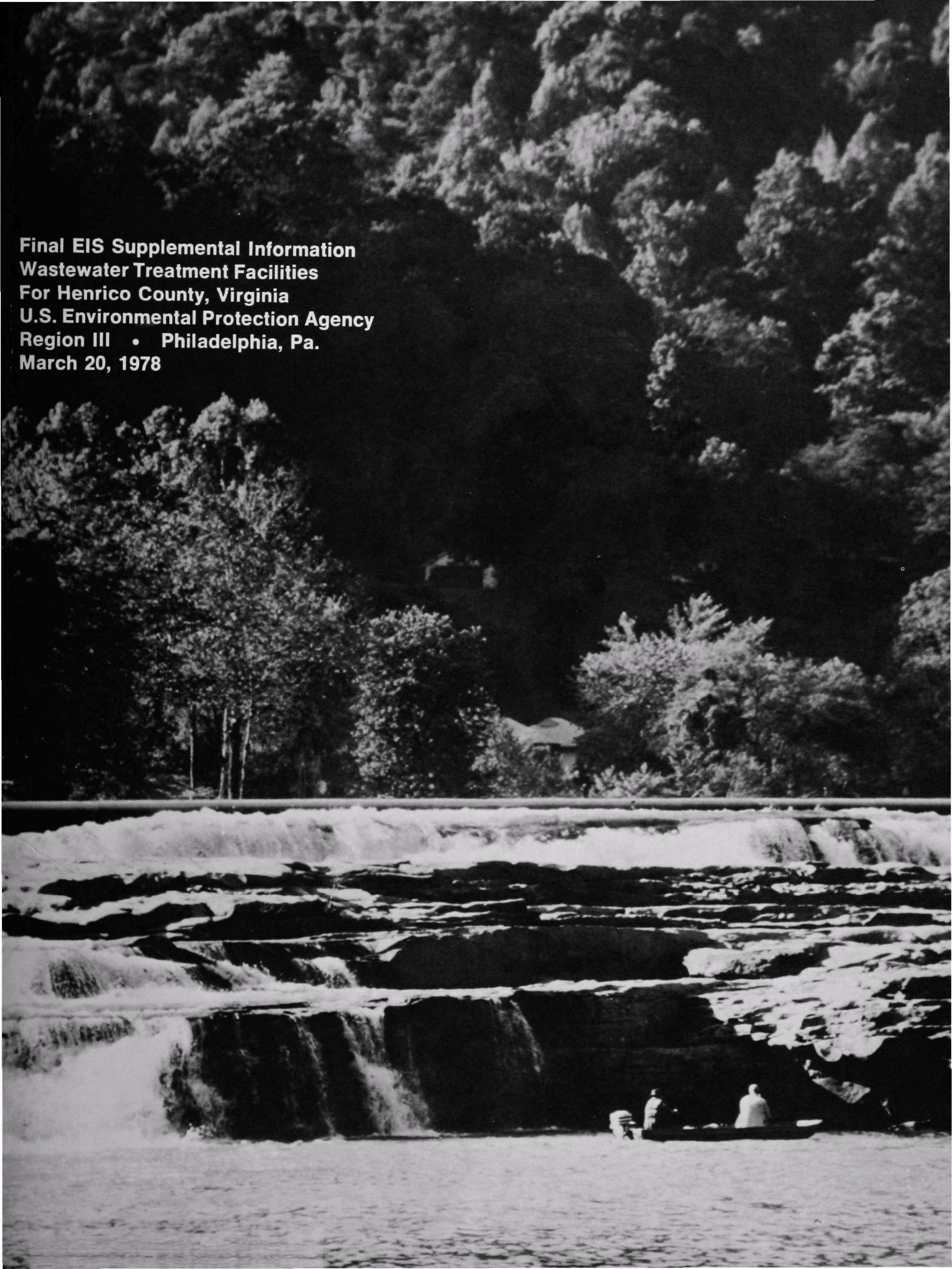
tration values are at stations upwind of the central urban area and the overestimated values at stations downwind of the urban area. The internal turbulent boundary layer can form when the surface conditions are altered by roughness or temperature changes (Shir, 1972b). This feature has not been included in the present study which essentially assumes the turbulent flows are in equilibrium with the surface conditions. As a consequence, the eddy diffusivity estimated at the upwind edge is too large and that estimated at the downwind edge too small. The result of this assumption leads to concentration under-estimates upwind of the urban area and overestimate downwind. More complicated effects due to the urban heat island, such as convection and non-uniform spatial stability, may also affect this discrepancy. However, effects on the stations which are far away from the central area are less. This can be seen in the good agreement of the observed data vs simulated values at stations 28, 33 and 36.

b. 24-hr averaged concentrations

Fig. 4 shows the 24-hr averaged time variation of the computed and observed concentration values. The observed values include 24-hr and 2-hr sampling data, the former being lower than the latter. This discrepancy makes comparison very difficult. The computed 24-hr mean concentrations shown in Fig. 4 follow the trend of the observed data. The agreement between the computed and observed data is very good at stations 36, 28, 12 and 33. The model computations underestimated the concentrations on the 4th, 5th, 9th, 14th and 20th of the month. This may be due to underestimated emission rates from the emission inventory model. The days on which overestimates occurred at stations 3 and 23 are for the northwest wind condition. The wind direction results in pollutant concentration being transported from large upwind emission sources. However, the transport effects do not show up on the observed data at stations 3 and 23 which are dominated by the local emission rates. This raises some doubts both in the model computation and the representation of the station data. The street canyon effects of high-rise buildings may prevent upwind concentrations from reaching the station. This causes strong local influences on the observed data. The strong microscale variations are even more pronounced for automobile exhaust. Hence the selections of station sites and the real representation of the observed data are quite important. On the other hand, the computed results can only represent mean concentration over a grid size which neglects microscale variations within that grid. The detailed microscale concentration distribution can only be resolved with a subgrid model such as that of Johnson *et al.* (1970). Such a subsystem for this model requires further developments. The excellent results at stations 36, 28 and 33 are apparently free from the effects of horizontal inhomogeneity and local influences, since these stations are located in relatively low emission areas far from the central region. The concentrations at these stations vary by two orders of magnitude depending on wind direction. These large concentration fluctuations are due to advection and are reproduced quite well by the model. The two sets of measured data, *viz.* 2- and 24-hr data, are also shown in Fig. 4. The 24-hr data are usually lower than the 2-hr data. On the 4th, 20th and 21st, the computed values at station 33 agree well with

TABLE 2. Correlation coefficients of computed vs observed 24-h averaged SO_2 concentrations.

Station no.	Objective analyzed wind field		Turner's analyzed wind field	
	Linear scale	Log scale	Linear scale	Log scale
3	0.380	0.334	0.398	0.325
15	0.617	0.598	0.635	0.620
17	0.363	0.466	0.408	0.502
23	0.471	0.647	0.444	0.642
33	0.832	0.725	0.830	0.718
4	0.177	0.284	0.220	0.313
10	0.139	0.145	0.105	0.125
12	0.830	0.821	0.812	0.804
28	0.828	0.886	0.823	0.878
36	0.914	0.910	0.937	0.940
Total	0.654	0.806	0.659	0.809



Final EIS Supplemental Information
Wastewater Treatment Facilities
For Henrico County, Virginia
U.S. Environmental Protection Agency
Region III • Philadelphia, Pa.
March 20, 1978

TABLE 3. Correlation coefficients of computed vs observed 2-hr averaged SO_2 concentrations.

Station no.	Numbers of cases	Objective analyzed wind field: Linear scale	Objective analyzed wind field: Log scale	Turner's analyzed wind field: Linear scale	Turner's analyzed wind field: Log scale
3	295	0.328	0.322	0.301	0.295
15	232	0.485	0.443	0.479	0.444
17	290	0.440	0.459	0.425	0.450
23	267	0.486	0.591	0.468	0.584
33	235	0.588	0.680	0.586	0.677
4	268	0.187	0.438	0.195	0.435
10	281	0.230	0.250	0.231	0.247
12	275	0.648	0.686	0.629	0.663
28	282	0.639	0.730	0.626	0.732
36	286	0.617	0.769	0.605	0.782
Total	2711	0.531	0.706	0.525	0.704

results of this model, e.g., stations 36, 28 and 12, follow closely the trend of observed data, although the observed data do show a highly fluctuating pattern. Hence, the use of this model for short-term pollution prediction is feasible.

For northwest wind conditions which occurred on the 1st, 3rd, 7th, 12th, 16th, 21st and 25th, the model underestimated the concentration at stations 3 and 23. As has been previously mentioned, the advection effects on the concentration do not appear on the observed data. Similar underestimated results occurred for the south wind conditions at most of the stations for the 5th, 6th, 9th, 14th and 19th days. It was found that high temperatures are associated with these days. The cause of underestimated results on these days is not completely clear. However, south wind conditions during our simulation period generally follow warm front passages. During such passage, emission may be underestimated due to unseasonable warm air. Also, the air motion and atmospheric stability are more complicated than has been specified in the model. Quite often, the computed trends have a phase shift from the observed data whenever air temperatures change rapidly. This indicates that the emission rates may have a certain time lag when responding to air temperature changes. The exact duration of the time lag seems to be quite complex and requires further investigation.

In general, the trend pattern of computed values follows reasonably close to that of the observed data. This is an important consideration on the feasibility of the model for short-term pollution prediction.

A few of the peak concentrations which appear on certain stations in Fig. 6 are mainly due to the influences of the large industrial sources near Alton, located 30 km north of St. Louis. These occur under north wind conditions. This phenomenon is strongly evident on 3 and 16 February, on the morning of 19 February at 22 stations, and somewhat weakly at noon on 4 and all February, and on the morning of 24 February at east-side (even number) stations. Thus, those large industrial sources contribute significantly to concentrations under this wind condition.

The correlation coefficients between the computed values and the observed data at ten stations are shown in Table 3. The correlation for all 10 stations is 0.531 in linear scale and 0.706 in log scale. The correlation based on Turner's analyzed wind field, which is also shown on Table 3, is very close to that based on objective analyzed wind. This is a 52% improvement from the Gaussian plume model which has a correlation of 0.347 based on regression line.

d. The frequency distribution of concentrations

The frequency distributions of combined 2-hr data at nine stations (excluding station 10) for different wind directions are shown in Fig. 7. The agreement is excellent for NE and SE wind directions. The maximum discrepancy in the concentration values between the 20 and 80 percentiles is 10 and 20 $\mu g\ m^{-3}$ for NE and SE wind conditions, respectively. The computed results overestimated the concentrations for NW winds and underestimated them for SW winds. These discrepancies

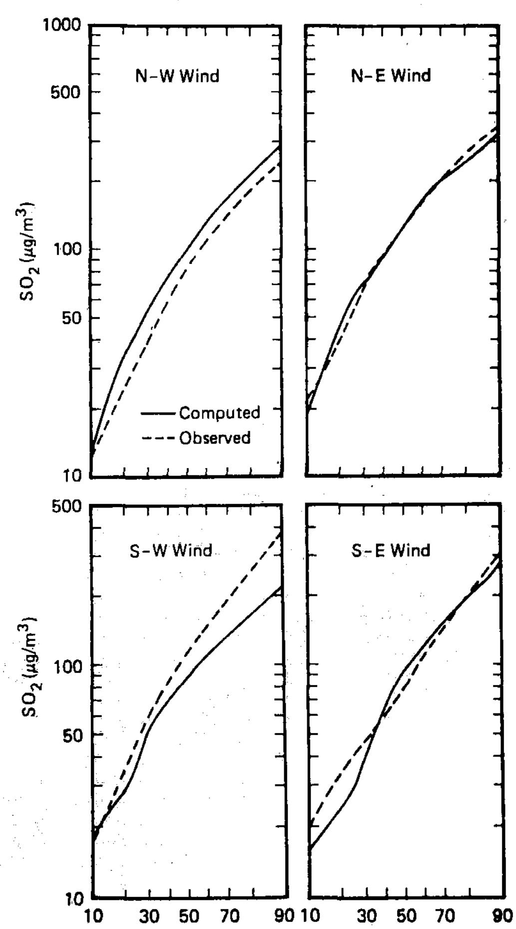


FIG. 7. Comparison of observed and computed 2-hr averaged frequency distribution of SO_2 concentration according to wind sectors (combined data from nine stations).

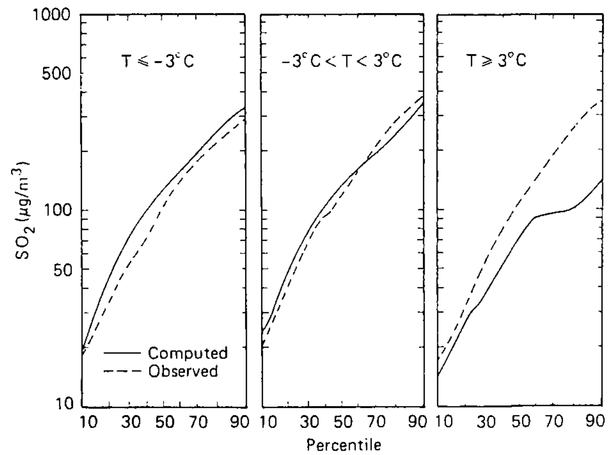


FIG. 8. As in Fig. 7 except for classification according to temperature range.

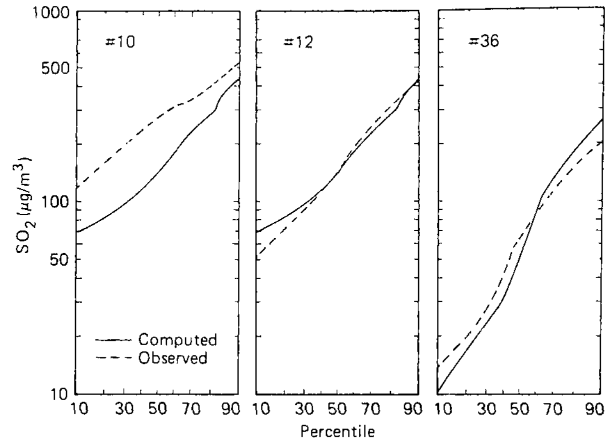


FIG. 9. Comparison of observed and computed 2-hr average frequency distributions of SO_2 concentration for three stations.

are consistent with the previous discussion. Since NW and SW winds often carry cold and warm air, respectively, these discrepancies may well be due to lag response of the emission rates to changing air temperatures.

This can be seen more clearly from Fig. 8 in which the frequency distributions for different temperature ranges are shown. The agreement is good when the air temperatures are between −3C and 3C. Overestimation and underestimation are seen when the temperatures are lower than −3C and higher than 3C, respectively.

Fig. 9 shows the frequency distributions at stations 10, 12 and 36, with good agreement at stations 12 and 36, and large underestimates at station 10 due to its proximity to point sources. The performance of the model at all ten stations is consistent with respect to the different statistical analyses. This reveals the possibilities of further improvements.

e. *Spatial distributions of surface concentrations*

In Fig. 10, we present a series of noteworthy cases which show the distribution of surface SO_2 concentrations under the specified conditions. A summary of these cases is given in Table 4.

Fig. 10a shows the surface SO_2 concentration field for a typical west wind condition in the St. Louis area. Atmosphere stability is nearly neutral. On this day (2 February 1965), the meteorological conditions were quite persistent. The source emission pattern is distinctly marked by the morning and evening hour peak concentrations, as reported by most of the stations (see Fig. 6), and simulated by the model. This figure shows the concentration field during the morning hour peak. The distribution pattern clearly indicates that there are four major axes of local maximum concentrations. These local high concentrations were resulted from the strong source emissions in the area of Alton, Granite City, East St. Louis and south St. Louis. The significant contributions of these sources to the surface concentrations can also be seen from the rest of the Figs. (10b–10i). The elongation of the closed isopleths is parallel to the mean wind direction. The local Alton area maximum (northeast of St. Louis) shows a simpler pattern than in the St. Louis and East St. Louis areas.

TABLE 4. Summary of various parameters used in the simulation of SO_2 concentrations shown in Figs. 10 and 11.*

Fig. no.	Day/Hour**	dd	ff	S	H	Q_A	Q_p	T	Remarks
10a	2/8	289	4.9	0.3	500	88547	10838	−16	typical west wind
10b	25/10	296	9.8	0	773	8140	11825	−12	high west wind
10c	13/4	302	2.2	1.9	710	4593	7123	−4	low west wind and stable atmosphere
10d	9/2	76	4.0	0.6	270	2732	6023	2	east wind
10e	8/14	19	4.0	−0.7	813	4835	10256	0	northeast wind, effects of emission sources at Alton area
10f	22/12	274	2.7	−1.8	942	6249	10899	−7	rotating wind, unstable atmosphere
10g	13/18	72	4.5	0.4	1000	4149	9332	0	rotating wind, stable atmosphere
10h	7/24	189	3.1	0.4	369	2115	6293	12	southwest wind, before cold front passage
10i	8/2	354	5.8	0.1	300	1682	6161	7	after cold front passage
11a	2/19	292	4.9	0.25	540	8570	12102	−15	typical west wind
11b	13/3	302	2.2	2.2	700	4593	7123	−4	
11c	3/8	358	3.1	0.3	400	7530	10517	−10	north wind

* dd, wind direction in degrees; ff, wind speed (m sec^{-1}); S, stability index; H, mixing height (m); Q_A, area source strength (gm sec^{-1}); Q_p, point source strength (gm sec^{-1}); T, temperature (°C).
** February 1965/LST.

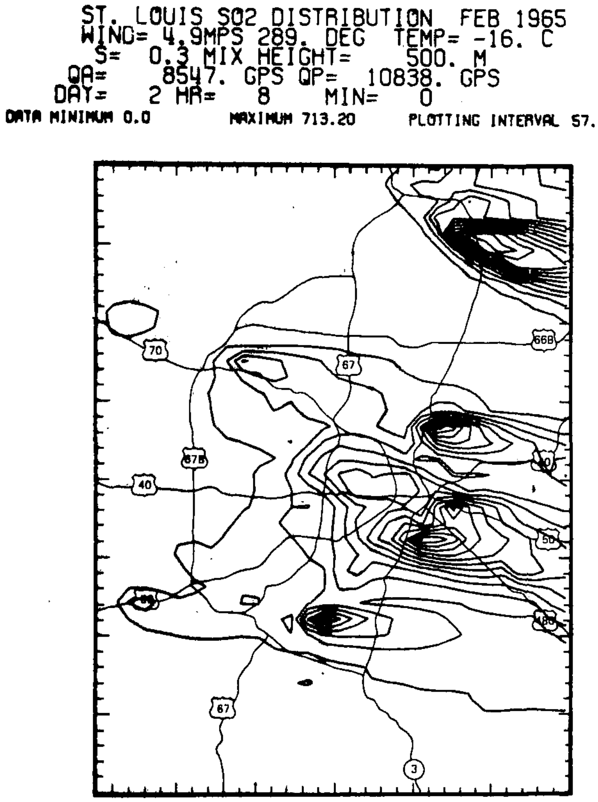


FIG. 10a. Simulation of 2-hr average surface SO_2 concentration field.

The effect of multiple emissions in the central urban area is prominent. The effect of wind speed on the pattern of the surface SO_2 concentration field is shown in Fig. 10b. The total source emission and the atmospheric stability are nearly the same as in the previous case. The increase in the wind speed results in a somewhat similar but simpler distribution pattern. The surface concentrations in the central area are reduced by about half, with the wind speed doubled from the previous case. This might suggest that the surface concentrations in the central area are approximately inversely proportional to the wind speed under the same prevailing meteorological conditions. However, as will be seen later, it is not that simple. The surface concentrations near Alton, where the point sources are dominant, are only reduced by 30%. This may be due to different plume rise: the increase of wind speed decreases the plume rise which in turn increases the surface concentrations. This phenomenon can be seen more clearly in Fig. 10c which shows the concentration pattern under low winds and a stable atmosphere. Although the total source emission is about 60% of that in the previous cases, the concentration pattern is quite different. The surface concentrations near Alton are largely reduced. There are two reasons for this—large plume rise due to low wind speed; and weak diffusive mixing, due to the stable atmosphere, which causes SO_2 to reach the ground slowly. The same reasons also explain the

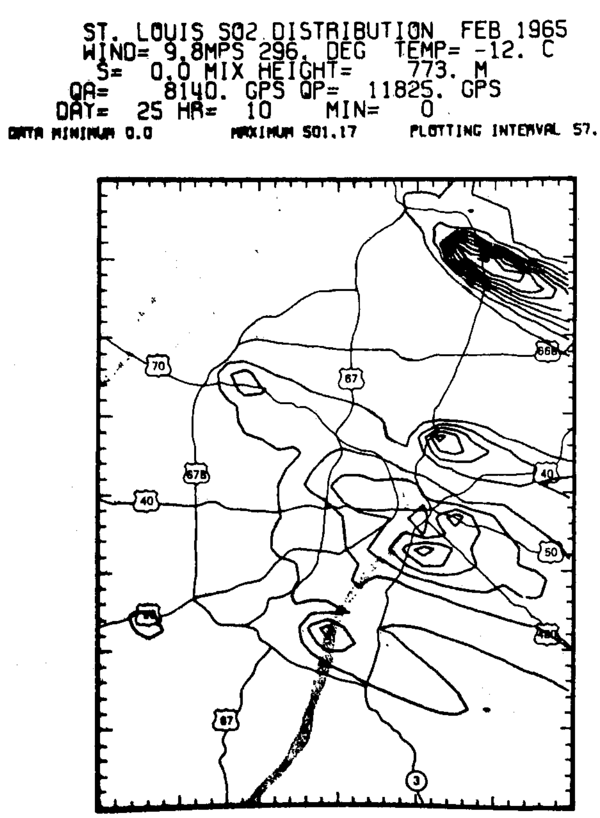


FIG. 10b. Simulation of 2-hr average surface SO_2 concentration field.

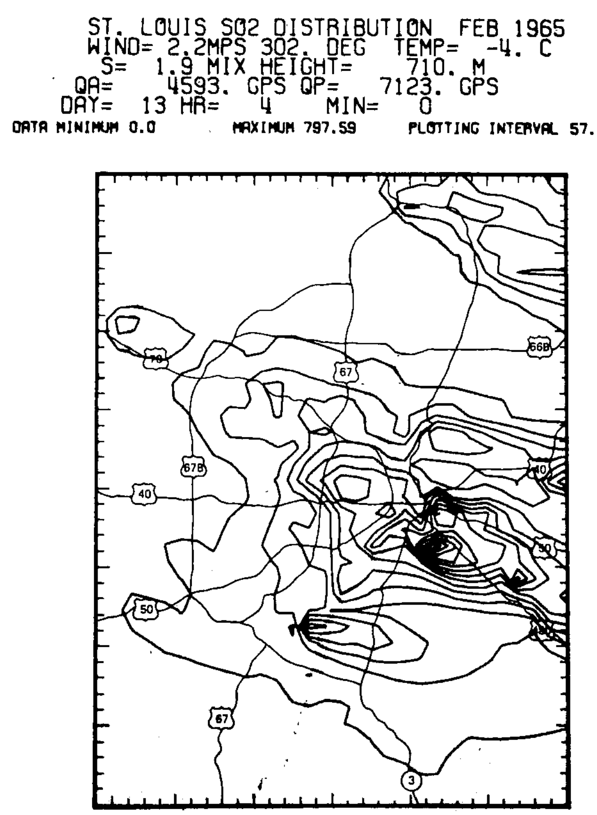


FIG. 10c. Simulation of 2-hr average surface SO_2 concentration field.

diminution of the local high concentration near Granite City. The local maximum concentration still occurs at East St. Louis, but is displaced somewhat eastward. The concentration levels at both St. Louis and East St. Louis are of equivalent intensity as in the 2 February case (Fig. 10a) even though source emissions have been reduced 60%. From these three figures (10a–c), as has been discussed, there is no simple relation between the surface concentrations and the local source-wind ratio, $Q/\bar{u}$, in these areas under the same wind direction. Thus, even under the same persistent wind direction, the variation of the wind speed can alter the concentration pattern significantly.

We now investigate the effects of the wind direction upon the concentration distributions. The next four figures show the different concentration patterns which result from various component winds; that for an east wind is given in Fig. 10d. The four major local maxima appear clearly (as in the west-wind pattern of Fig. 10a), except that the locations of the maxima are shifted westward. This figure clearly demonstrates that the variations of wind direction can change the concentration pattern completely. Thus, as has been discussed already, the concentration distributions are greatly influenced by atmospheric stability, wind speed and direction orientation with respect to the source distributions. Moreover, the concentrations in the downtown

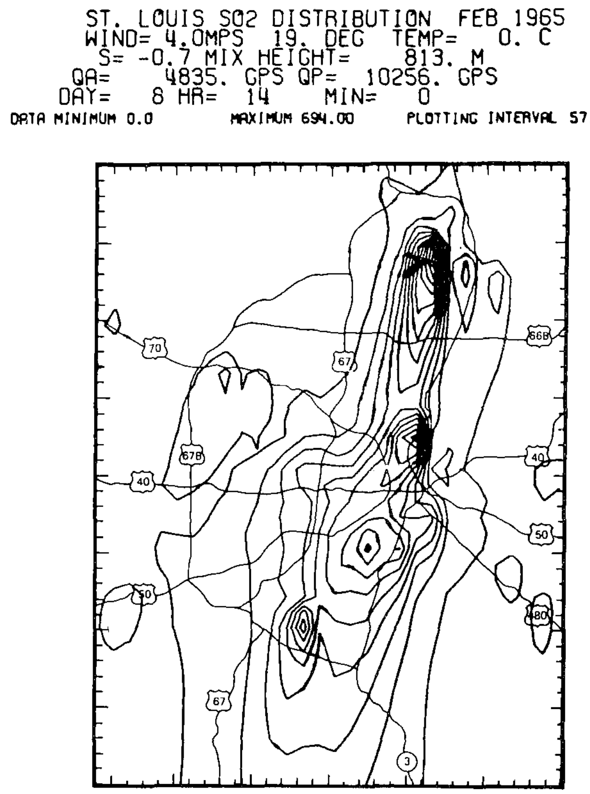


FIG. 10c. Simulation of 2-hr average surface SO_2 concentration field.

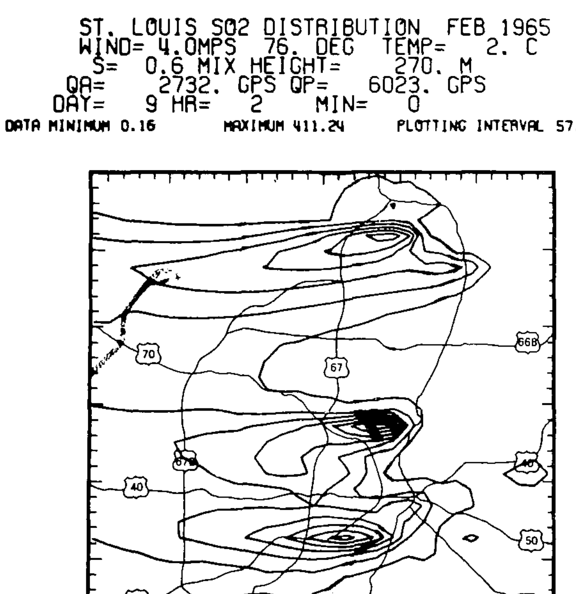


FIG. 10d. Simulation of 2-hr average surface SO_2 concentration field.

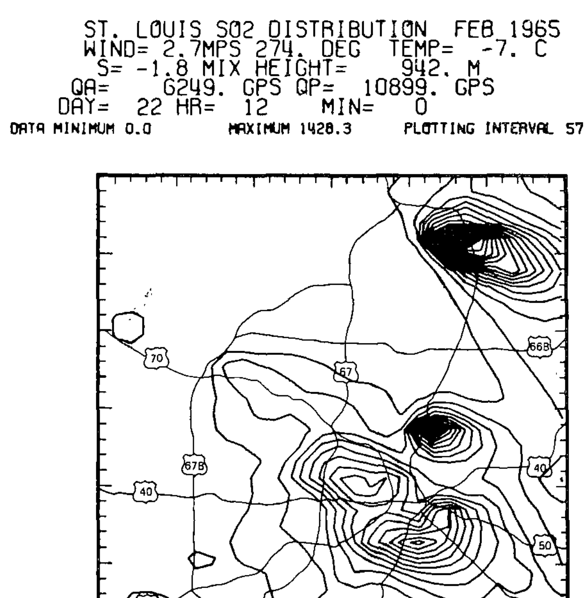


FIG. 10f. Simulation of 2-hr average surface SO_2 concentration field.

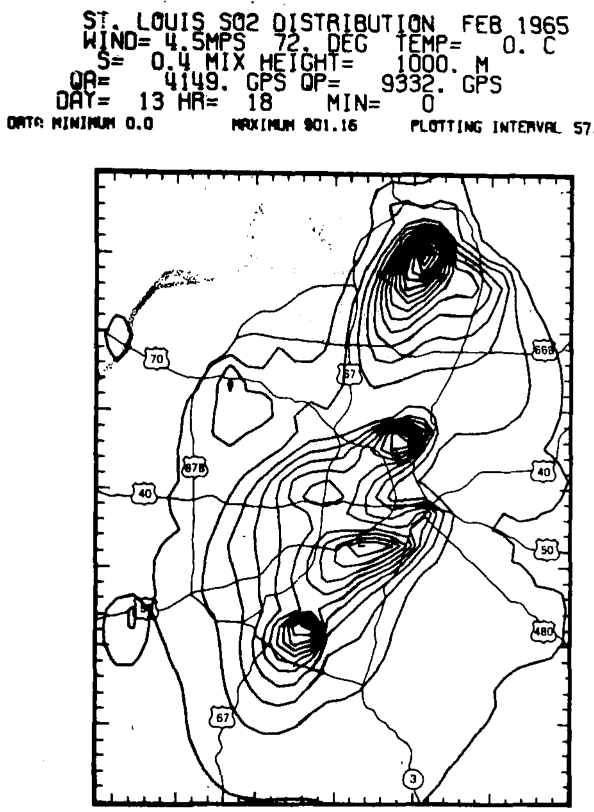


FIG. 10g. Simulation of 2-hr average surface SO_2 concentration field.

St. Louis area usually have greater values during easterly than during westerly winds.

A case of no morning and evening peak concentrations was reported on 8 February. Instead, a single peak concentration appears at noon (Fig. 6). This phenomenon is more pronounced for the station located on the west side of the Mississippi River; the concentration field is reproduced in Fig. 10e. Throughout this day, the wind speed and source emission rate were held quite constant, and the temporal variation in stability was also small. The reason for this single peak concentration in the daily variation can be attributed to two factors. One is the rotation of wind direction from NW in the morning to NE at noon. This tends to make pollutants in the air remain in the urban area. The major contributing factor was that the pollutants emitted from the industrial area (located at Alton) were transported to the metropolitan center. For all wind directions, downtown St. Louis has the highest concentration values under the northeast winds. Fig. 10e clearly explains the reason. The contribution of industrial emissions to the concentration in the metropolitan area were also reported for 3 February. However, during this day, the wind changed direction from west to north, then back to west in the morning and repeated the same pattern in the afternoon. The north wind condition coincided with morning and evening peak source emissions. The observed concentration

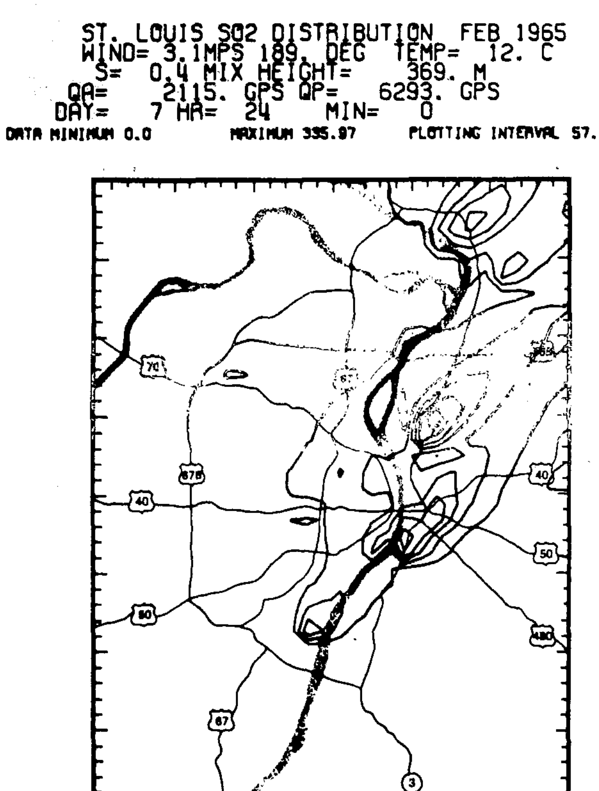


FIG. 10h. Simulation of 2-hr average surface SO_2 concentration field.

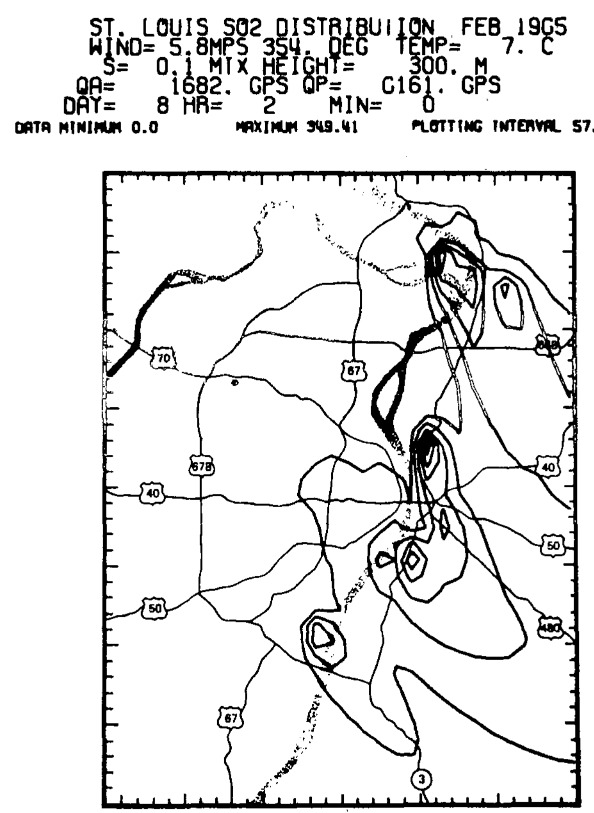


FIG. 10i. Simulation of 2-hr average surface SO_2 concentration field.

values at most stations showed sharp peaks during these hours.

Low wind speeds associated with a rapidly rotating wind direction will have the effect of distributing pollutants in a more circular pattern. This happened on 13 and 22 February. On these two days the wind changed direction more than 180° in a very short period; the resulting circular trajectory of the air had a dramatic effect on the transport of the pollutants in the atmosphere. This is illustrated in Figs. 10f and 10g for 22 and 13 February, respectively. The former is for an unstable atmosphere and the latter for a slightly stable atmosphere. Apparently, atmospheric stability has little effect on the distribution pattern when the wind direction is changing rapidly. However, one still can notice the effect of final wind direction on these two figures. Our experience indicates that a steady-state model is not able to reproduce the resultant concentration fields shown in Figs. 10f and 10g. It should be noted that concentration values show a great temporal variation from one station to another due to the orientation of source-receptor locations and its relationship with respect to wind direction. In some cases, a different trend is reported [e.g., stations 28 and 36 on 22 February (Fig. 6)]. The comparison of computed concentrations and observed data during these periods is remarkable. This indicates that model computation is able to respond to fast changing atmospheric conditions. Thus, the concentration distributions also depend on the history of the wind.

A cold front passed the St. Louis Metropolitan area at 2400 LST 7 February. Figs. 10h and 10i show the concentration fields before and after the frontal passage, respectively. In a 2-hr period, the wind changed its direction from southwest to northwest and picked up speed. In the same period the ambient temperature dropped from 12 to 7C, and eventually reached 0C. All stations reported a sharp decrease in concentration during this period followed by rapid increase after frontal passage (Fig. 6). The model is able to predict this trend pattern. However, there is a slight time lag in the observed minimum as compared with the computed value. The reason is quite complicated. We tend to agree that the disparity between the analyzed and actual wind fields probably is a contributing factor. Fig. 10h also represents a typical concentration field for the St. Louis area under southwest winds.

f. Vertical distributions of concentrations

Very few up-to-date investigations have been made to understand the vertical distribution of pollutants in an urban atmosphere. While our model computes the three-dimensional distributions of SO_2 concentration, unfortunately, there are no observational data to verify its results. The following discussions are some phenomenological conclusions indicated by the computational results.

Figs. 11a–c show vertical cross sections (x-z plane) of SO_2 concentrations.

The chosen cross section which is located 19 source grids from the bottom (see Fig. 1) passes through downtown St. Louis. A summary of these cases is included in Table 4. Fig. 11a represents the vertical concentration pattern parallel to the wind for medium-strength west winds and an unstable atmosphere. Two distinct concentration maxima occurred east of the St. Louis and East St. Louis areas. Large amounts of SO_2 were dispersed up to the mixing height due to high turbulent mixing under unstable conditions. The slopes of the isopleths at downtown St. Louis are larger than those at rural areas due to large urban roughness. The high concentration and large slopes of the isopleths at East St. Louis were caused by the elevated industrial sources. Fig. 11b is for a case of low west winds with a stable atmosphere. Even with lower winds the slopes of the isopleths are much smaller than those in the previous

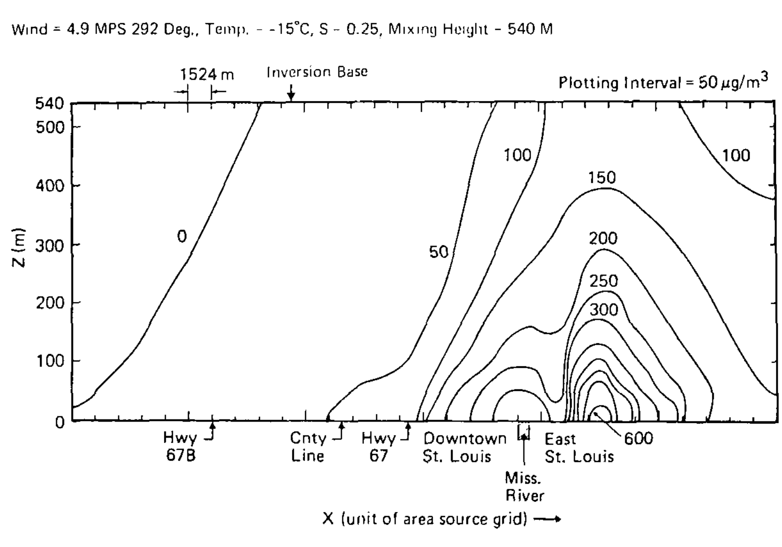


FIG. 11a. Simulation of hourly averaged SO_2 concentration field in the x-z plane.

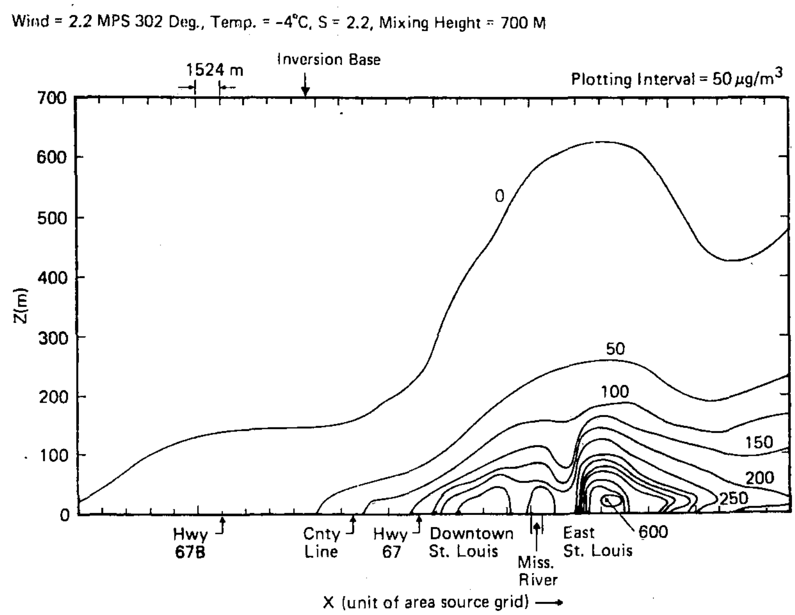


FIG. 11b. Simulation of hourly averaged SO_2 concentration field in the x-z plane.

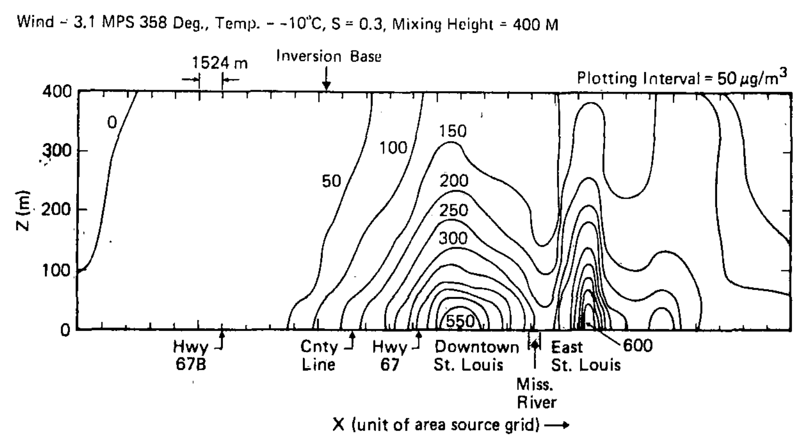


FIG. 11c. Simulation of hourly averaged SO_2 concentration field in the x-z plane.

case due to low mixing under stable conditions. Moreover, the concentration distributions in both cases are not similar, even though they both are for the same wind direction. The local concentration maxima occur on the surface in the former case but not in the latter. The elevated local maximum at East St. Louis in the latter case is due to low mixing in the stable air flow; thus, the plumes from the elevated sources are dispersed slowly. Thus, the vertical concentration distribution along the wind direction is influenced by roughness, stability, and the source position. Fig. 11c shows the crosswind concentration pattern under low north winds and unstable atmospheric conditions. The local concentration maximum was displaced westward due to the different wind direction. The concentrations at downtown St. Louis are largely increased due to the low mixing height and unstable conditions.

From these three figures, it is clear that the vertical concentration distribution of pollutants is neither uniform nor similar in the mixing layer. The hypothesis of constant concentration in the vertical is only achieved (approximately) at a far distance from major source emissions and under unstable atmospheric conditions. The slope of the isopleths on the upwind edge is related to atmospheric stability. Horizontal wind speed has no appreciable effect. Other parameters that might affect this slope are change in surface roughness and spatial distribution of source emission. These figures indicate that the vertical concentration distributions are as irregular as horizontal concentration distributions. Thus, no obvious simplification can be made. Such an attempt requires further investigation.

6. Conclusions and recommendations

The findings of this investigation provide some insight into the relationship between concentrations and sources and their dependence on various parameters. The results also indicate the capabilities and limitations of the modelling approach. The main attractiveness of this model is its rather consistent performance under the various conditions which occurred within the 25-day test period. Consistent performance of the model was especially satisfactory for both strong and light wind conditions. Similar performance was noted for sudden changes of wind directions. This is an improvement over the Gaussian plume model tested for the same time period. In addition, its major advantages are a large flexibility in handling cases with arbitrary distributions of sources with varying emission rates, with spatial and temporal variations of wind (including vertical wind), eddy diffusion coefficients, stability, and mixing height. It also can deal with spatially varying surface roughness and topographical conditions as well as nonlinear chemical reactions. With growing availability of data and the advance in the understanding of urban meteorology, atmospheric turbulence and chemical reactions of pollutants, the full potential of the model will be more appreciated in the near future.

On the other hand, its current disadvantages are those of neglecting microscale variations of pollutants from concentrated sources, and the detailed urban effects upon the pollutant dispersion. The shortcomings of the numerical method are the constraints of the integration time step required by the stability, accuracy criteria, and the inevitable numerical errors. The latter does not appear to be a severe limitation. The newly developed methods (Gazdag, 1972; Orszag, 1971) can provide very accurate numerical results. This method can be applied whenever the accuracy requirement is such that the errors from the current method can no longer be tolerated. More efforts are needed in grid resolution improvement by introducing a subgrid model.

In addition, the effect of the urban area on creating a non-equilibrium atmospheric turbulent structure needs further investigation. This kind of study may give more information on the turbulence diffusion, vertical air motion, and spatial variation of atmospheric stability and mixing height over the urban area. Improvements on the input data are also necessary. The estimated diurnal varying emission rates which are governed by the ambient air temperature seem to be inaccurate during the early periods of unseasonably warm or cold spells. Improvement may be made by considering the effects of the history of the air temperature upon the emission rates. The site of the monitoring station should be selected carefully so that the measurements are free from local influences and therefore provide a better representation of the ambient air quality. The applicability of the bulk Richardson number and the horizontal wind fluctuations to determine the atmospheric stability requires further study. With respect to the procedures for handling the vast amounts of input and output data, some auxiliary programs are needed. Continuous graphic displays of the data and various statistics are very useful in understanding the complex phenomena which result from such an immense amount of data. After all, any attempt at research modeling should first aim toward understanding the overall phenomenon rather than emphasizing the simplicity and convenience of the model.

Acknowledgments. The authors sincerely thank Messrs. Bruce Turner, John Zimmerman and Dr. James L. McElroy of EPA for their cooperation in providing the basic data in this study. The authors are grateful to Mr. Paul K. Halpern and Drs. John A. Barker and William E. Langlois for proofreading the manuscript.

REFERENCES

Businger, J. A., *et al.*, 1971: Flux-profile relationships in the atmospheric surface layer. *J. Atmos. Sci.*, **28**, 181–189.

Farmer, J. R., and J. D. Williams, 1966: Air quality measurement: Interstate air pollution study, Phase II, Vol. III, of Project Report. National Center for Air Pollution Control, PHS, Cincinnati, Ohio.

Frenkiel, F. N., 1956: Atmospheric pollution and zoning in an urban area. *Sci. Monthly*, **82**, No. 4, 194–203.

Gazdag, J., 1972: Numerical convective schemes based on accurate space derivatives. Sci. Rept. 320–3305, IBM Scientific Center, Palo Alto, Calif.

Gifford, F. A., Jr., and S. R. Hanna, 1970: Urban air pollution modeling. Paper ME-32D presented at Second Intern. Clean Air Congress, IUAPPA.

Golder, D., 1972: Relations among stability parameters in the surface layer. *Boundary-Layer Meteor.*, **3**, 47–57.

Harrington, B., Jr., 1965: Atmospheric diffusion of ragweed pollutant in urban area. Final Report, Grant AP 00006, University of Michigan, Ann Arbor.

Iordanov, D. L., 1966: On diffusion from a point source in the atmospheric surface layer. *Izv. Atmos. Oceanic Phys.*, **2**, 576–584.

Johnson, W. B., F. L. Ludwig and A. E. Moon, 1970: Development of a practical multi-purpose urban diffusion model for carbon monoxide. *Proc. Symp. Multiple-Source Urban Diffusion Models*, A. C. Stein, Ed., APCO Publ. No. AP-86, Research Triangle Park, N. C.

Lamb, R. G., and M. Neiburger, 1971: An interim version of a generalized urban air pollution model. *Atmos. Environ.*, **5**, 239–264.

Leahey, D. M., 1972: An advective model for predicting air pollution within an urban heat island with application to New York City. *J. Air Pollut. Control Assoc.*, **22**, 548–555.

Leighton, P. A., 1961: *Photochemistry of Air Pollution*. New York, Academic Press, 300 pp.

Lettau, H. H., 1970: Physical and meteorological basis for mathematical models of urban diffusion processes. *Proc. Symp. Multiple-Source Urban Diffusion Models*, A. C. Stern, Ed., APCO Publ. No. AP-86, Research Triangle Park, N. C.

MacCracken, M. C., T. V. Crawford, K. R. Peterson and J. B. Knox, 1972: Initial application of a multi-box air pollution model to the San Francisco bay area. *Proc. Joint Automatic Control Conf.*, Stanford University, Stanford, Calif.

McElroy, J. L., and F. Pooler, Jr., 1968: St. Louis dispersion study, Vol. I—Instrumentation, Procedures, and Data Tabulations; Vol. II—Analysis. Nat. Air Pollut. Control Admin., PHS, Washington, D. C.

Moses, H., 1969: Mathematical urban air pollution models. Paper 69-31 presented at the 62nd Annual Meeting, Air Pollut. Control Assoc., Pittsburgh, Pa.

Neiburger, M., 1971: Meteorological aspects of air pollution and simulation models of diffusion, transport and reactions of air pollution. Project Clean Air Rept., Dept. of Meteorology, UCLA, Los Angeles.

Orszag, S. A., 1971: Accuracy of numerical simulation of incompressible flows. *J. Fluid Mech.*, **49**, 75–112.

Pasquill, F., 1962: *Atmospheric Diffusion*. London, Van Nostrand, 297 pp.

Pelton, D. J., *et al.*, 1972: Validation and sensitivity analysis of the Gaussian plume multiple-source urban diffusion model. APTD-0935, Office of Air Program, EPA, Research Triangle Park, N. C.

Randerson, D., 1970: A numerical experiment in simulating the transport of sulfur dioxide through the atmosphere. *Atmos. Environ.*, **4**, 615–632.

Reynolds, S. D., M. K. Liu, T. A. Hecht, P. M. Roth and J. H. Seinfeld, 1973: Further development and evaluation of a simulation model for estimating ground level concentrations of photochemical pollutants. Final Report, EPA Contract 68-02-0339, Systems Applications, Inc., Beverly Hills, Calif.

Ritchmyer, R. D., and K. W. Morton, 1967: *Difference Methods for Initial Value Problems*. New York, Interscience, 405 pp.

Roberts, J. J., E. J. Crake and A. S. Kennedy, 1970: An urban atmospheric dispersion model: *Proc. Symp. Multiple-Source Urban Diffusion Models*, A. C. Stern, Ed., APCO Publ. No. AP-86, Research Triangle Park, N. C.

Rounds, W., Jr., 1955: Solution of the two-dimensional diffusion equation. *Trans. Amer. Geophys. Union*, **36**, 395–405.

Shieh, L. J., P. K. Halpern, B. A. Clemens, H. H. Wang and F. F. Abraham, 1972: Air quality diffusion model: Application to New York City. *IBM J. Res. Develop.*, **16**, 162–170.

Shir, C. C., 1970: A pilot study in numerical techniques for predicting air pollutant distribution downwind from a stack. *Atmos. Environ.*, **4**, 387–407.

——, 1972a: Numerical investigation of the atmospheric dispersion of stack effluents. *IBM J. Res. Devel.*, **16**, 172–179.

——, 1972b: A numerical computation of air flow over a sudden change of surface roughness. *J. Atmos. Sci.*, **29**, 304–310.

——, 1973: A preliminary numerical study of atmospheric turbulent flows in the idealized planetary boundary layer. *J. Atmos. Sci.*, **30**, 1327–1339.

Sklarew, R. C. *et al.*, 1972: Mathematical modeling of photochemical smog using the PICK method. *J. Air Pollut. Control Assoc.*, **22**, 865–869.

Turner, D. B., 1964: A diffusion model for urban areas. *J. Appl. Meteor.*, **3**, 83–91.

——, 1968: The diurnal and day-to-day variations of fuel usage for space heating in St. Louis, Missouri. *Atmos. Environ.*, **2**, 339–351.

——, 1969: Workbook of atmospheric dispersion estimates. PHS Publ. No. 999-AP-26.

——, and N. G. Edmisten, 1968: St. Louis SO_2 dispersion model study—Description of basic data. Unpubl. report, Air Pollution Office, EPA, Durham, N. C.

Venezia, R., and G. Ozolin, 1966: Air pollutant emission inventory: Interstate Air Pollution Study, Phase II, Vol. II, of Project Report. National Center for Air Pollution Control, PHS, Cincinnati, Ohio.

Wendell, L. L., 1970: A preliminary examination of mesoscale wind fields and transport determined from a network of wind towers. NOAA Tech. Memo. ERLTM-ARL25, Silver Spring, Md.

Wilson, W. E., Jr., and A. Levy, 1968: A study of sulfur dioxide in photochemical Smog. Rept., Battelle Memorial Institute, Columbus, Ohio.

6. Finite Difference Scheme for the Horizontal Advection terms of the Concentration Equation.

Finite difference scheme for the horizontal advection terms of the concentration equation.

We shall consider an equation

$$\frac{\partial c}{\partial t} = - \frac{\partial uc}{\partial x} - \frac{\partial vc}{\partial y} \qquad \text{(A-1)}$$

and grids layout as shown in Figure A-1. The notation used in Part I of this report is retained in this section.

We propose that the solution of equation (A-1) is to be approximated by a second order, central finite difference scheme with the application of the flux corrections. A time splitting method* will be employed to compute two terms on the right-hand side of equation (A-1).

Refer to Figure A-1, the rate of change of the mass (concentration value c) at grid point (i,j) is depended on the fluxes cross through the edge surfaces at a, b, c and d. Note that the grid cell represented by the grid point (i,j) has the size of $\frac{1}{2}(\Delta x_i + \Delta x_{i-1})$ and $\frac{1}{2}(\Delta y_j + \Delta y_{j-1})$ in x and y direction respectively. The following discussions apply to non-uniform grid spacings.

*It is also called "fractional-step method" or "splitting-up method." It is a method to replace a multidimensional problem by a succession of one-dimensional problems. (See Chapters 3 and 4 of "Numerical Methods in Weather Prediction" by G. I. Marchuk, Academic Press, New York, 1974.)

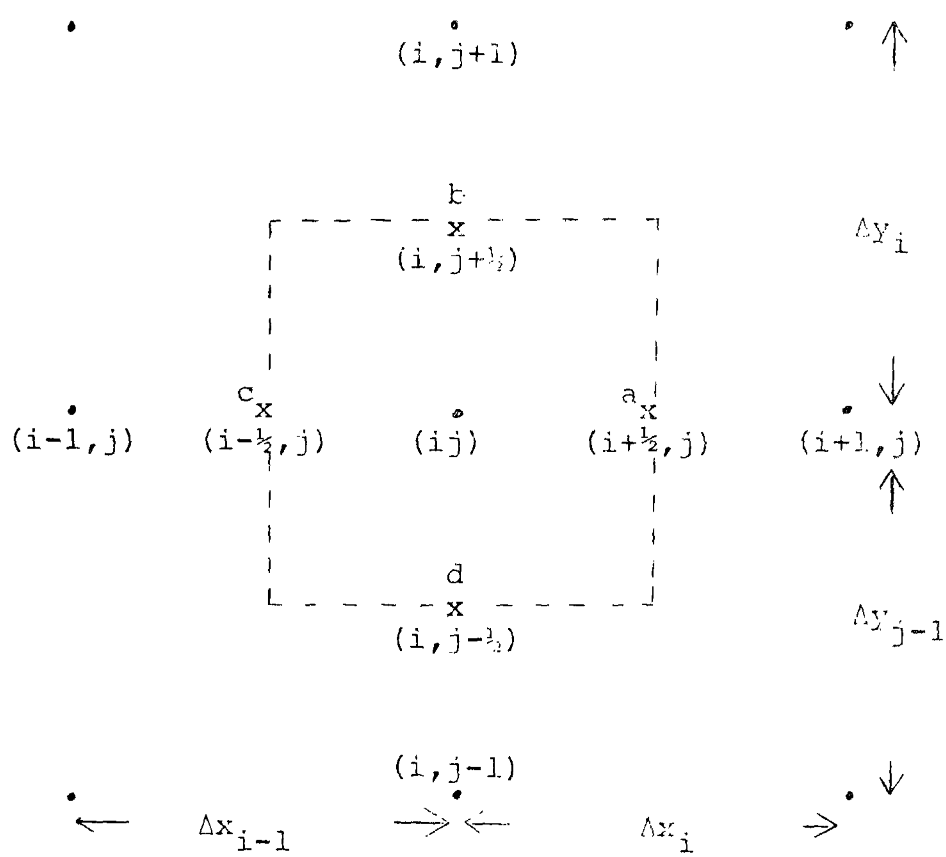


Figure A-1 Grids layout

Expand C with respect to time at grid points (i,j) in a Taylor's series, we have

$$C_{ij}^{n+1} = C_{ij}^{n} + \frac{\partial c}{\partial t}\bigg|_{ij} (\delta t) + \frac{\partial^2 c}{\partial t^2}\bigg|_{ij} \frac{(\delta t)^2}{2} + O\,(\delta t)^3 . \qquad \text{(A-2)}$$

Applying equation (A-1) and assuming u and v do not vary in a time step, (this is relevent to our problem because u and v change every hour) equation (A-2) becomes

$$\begin{aligned} C_{ij}^{n+1} = C_{ij}^{n} &+ \delta t \left(- \frac{\partial uc}{\partial x} - \frac{\partial vc}{\partial y}\right) \bigg|_{ij}^{n} \\ &+ \frac{(\delta t)^2}{2}\left(\frac{\partial}{\partial x}\, u\frac{\partial uc}{\partial x} + \frac{\partial}{\partial y}\, v\frac{\partial vc}{\partial y}\right) \bigg|_{ij}^{n} \\ &+ \frac{(\delta t)^2}{2}\left(\frac{\partial}{\partial x}\, u\frac{\partial vc}{\partial y} + \frac{\partial}{\partial y}\, v\frac{\partial vc}{\partial y}\right) \bigg|_{ij}^{n} , \qquad \text{(A-3)} \end{aligned}$$

or

$$C_{ij}^{n+1} = C_{ij}^{n} + \hat{U}[C_{ij}^{n}] + \hat{V}[C_{ij}^{n}] + \underbrace{\frac{(\delta t)^2}{2}\left(\frac{\partial}{\partial x}\, u\frac{\partial vc}{\partial y} + \frac{\partial}{\partial y}\, v\frac{\partial uc}{\partial x}\right)}_{\text{----Cross terms----}} \qquad \text{(A-4)}$$

where

$$\hat{U}[C_{ij}^{n}] = -\,\delta t\, \frac{\partial uc}{\partial x}\bigg|_{ij}^{n} + \frac{(\delta t)^2}{2}\left(\frac{\partial}{\partial x}\, u\, \frac{\partial uc}{\partial x}\right)\bigg|_{ij}^{n} ; \qquad \text{(A-5)}$$

$$\hat{V}[C_{ij}^{n}] = -\,\delta t\, \frac{\partial vc}{\partial y}\bigg|_{ij}^{n} + \frac{(\delta t)^2}{2}\left(\frac{\partial}{\partial y}\, v\, \frac{\partial vc}{\partial y}\right)\bigg|_{ij}^{n} \qquad \text{(A-6)}$$

The cross terms in equation (A-4) can be approximated by a time splitting method. If we assume that the computation of the new concentration value at grid point (i,j) is performed as follows:

$$C^{*}_{ij} = C^{n}_{ij} + \hat{U}[C^{n}_{ij}] \quad , \tag{A-7a}$$

and

$$C^{n+1}_{ij} = C^{*}_{ij} + \hat{V}[C^{*}_{ij}] \quad . \tag{A-7b}$$

It can be shown that

$$C^{n+1}_{ij} = C^{n}_{ij} + \hat{U}[C^{n}_{ij}] + \hat{V}[C^{n}_{ij}] + (\delta t)^2 \left(\frac{\partial}{\partial y} v \frac{\partial uc}{\partial x}\right)\Bigg|^{n}_{ij} + O[(\delta t)^3] \tag{A-8}$$

By reversing the order of computation of $\hat{U}$, $\hat{V}$ in equation (A-7), we have

$$C^{n+1}_{ij} = C^{n}_{ij} + \hat{U}[C^{n}_{ij}] + \hat{V}[C^{n}_{ij}] + (\delta t)^2 \left(\frac{\partial}{\partial x} u \frac{\partial vc}{\partial y}\right)\Bigg|^{n}_{ij} + O[(\delta t)^3] \tag{A-9}$$

Equations (A-8) or (A-9) shows that the computational method based on equations (A-7a) and (A-7b) is a good approximation for equation (A-4), if velocity field has a small spatial variation. Thus, equation (A-4) can be represented by

$$C_{ij}^{n+1} = C_{ij}^{n} + \hat{U}[C_{ij}^{n}] + \hat{V}[C_{ij}^{*}] \quad . \tag{A-10}$$

The finite difference approximation for equation (A-5) is to evaluate the fluxes across the edge surface at a and c shown in Figure (A-1). Thus,

$$\hat{U}[C_{ij}^{n}] = \Delta t\ [U_c(C_{i-1,j}^{n} + C_{ij}^{n})/2 - U_a(C_{ij}^{n} - C_{i+1,j}^{n})/2]/\tfrac{1}{2}(\Delta x_i + \Delta x_{i-1})$$
$$- \frac{(\Delta t)^2}{2}\ [U_c^2\ (C_{ij}^{n} - C_{i-1,j}^{n})/\Delta x_{i-1} - U_a^2\ (C_{i+1,j}^{n} - C_{ij}^{n})/\Delta x_i]$$
$$/\tfrac{1}{2}(\Delta x_i + \Delta x_{i-1}) \quad , \tag{A-11}$$

where $U_a = U_{i+\frac{1}{2},j}$ and $U_c = U_{i-\frac{1}{2},j}$. By applying equation (18c) in Section I-2-C-b, equation (A-11) can be rewritten as

$$\hat{U}[C_{ij}^{n}] = \hat{F}_x[C_{ij}^{n}] - \hat{F}_x^{*}[C_{ij}^{n}] \quad , \tag{A-12}$$

where $\hat{F}_x[C_{ij}^{n}]$ and $\hat{F}_x^{*}[C_{ij}^{n}]$ have the same expressions as equations (18a) and (18b) in Section I-2-C-b. Similarly, the finite difference approximation for equation (A-6) is

$$\hat{V}[C_{ij}^{n}] = \hat{F}_y[C_{ij}^{n}] - \hat{F}_y^{*}[C_{ij}^{n}] \quad . \tag{A-13}$$

The flux correction method is applied to compute equations (A-12) and (A-13). For x direction advection, the flux correction method is defined as

$$C_{ij}^{n+1} = C_{ij}^{n} + \mathcal{U}[C_{ij}^{n}] \tag{A-14}$$

where

$$\mathcal{U}[C_{ij}^{n}] = F_x[C_{ij}^{n}] - F'_x\,[C_{ij}^{n}] \ , \tag{A-15}$$

and

$$\begin{aligned} F_x &= \min\,(C_{i-1,j}\ ,\ \hat{F}_x)\ ,\quad \text{if } \alpha_{xi} > 0\ ; \\ &= \min\,(C_{ij}\ ,\ \hat{F}_x)\ ,\quad \text{if } \alpha_{xi} < 0\ , \end{aligned} \tag{A-16a}$$

$$\begin{aligned} F'_x &= \min\,(C_{ij}\ ,\ \hat{F}'_x)\ ,\quad \text{if } \alpha_x{}^{*}{}_{i} > 0\ ; \\ &= \min\,(C_{i+1,j}\ ,\ \hat{F}'_x)\ ,\quad \text{if } \alpha_x{}^{*}{}_{i} < 0\ . \end{aligned} \tag{A-16b}$$

Similar treatment is applied to y direction advection. Thus the finite difference formula for equation (A-1) is

$$C_{ij}^{n+1} = C_{ij}^{n} + \mathcal{U}[C_{ij}^{n}] + \mathcal{V}[C_{ij}^{*}] \tag{A-17}$$

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA 600/4-75-005b	2.	3. RECIPIENT'S ACCESSION NO.
4. TITLE AND SUBTITLE DEVELOPMENT OF AN URBAN AIR QUALITY SIMULATION MODEL WITH COMPATIBLE RAPS DATA: VOLUME II		5. REPORT DATE May 1975 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) C. C. Shir and L. J. Shieh		8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS IBM Research Laboratory San Jose California 95193 IBM Scientific Center, Palo Alto, Calif. 94304		10. PROGRAM ELEMENT NO. 1AA003 11. CONTRACT/GRANT NO. 68-02-1833
12. SPONSORING AGENCY NAME AND ADDRESS Environmental Protection Agency Environmental Research Center Research Triangle Park, N.C. 27711		13. TYPE OF REPORT AND PERIOD COVERED Final Report 7/1/74-5/30/75 14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Issued as Volume II of 2 Volumes

16. ABSTRACT

An advanced generalized urban air quality model (IBMAQ-2) is developed based on the theory utilized in an existing model (IBMAQ-1) as prescribed in Ref. 1. The model, based on numerical integration of the concentration equation, computes temporal and three-dimensional spatial concentration distributions resulting from specified urban point and area sources by using NEDS (National Emission Data System) and simulated RAMS (Regional Air Monitoring System) data. The UTM (Universal Transverse Metric) coordinates are used in all geographical, source emission, and monitoring data. A new menthod to incorporate point sources into the grid computation is developed by using a Lagrange trajectory method. Many model options are provided w which enable users to study conveniently the significant effects which these options have on the final concentration distribution

The program description is included to provided a guide for users. The program is constructed in a modular form which allows users to change or improve each componet conveniently. The input auxiliary model, which processes geographical, source emission, and monitoring data, is alos included.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
Air Pollution Boundary Layer Modeling Mathematical Modeling		

18. DISTRIBUTION STATEMENT UNLIMITED	19. SECURITY CLASS *(This Report)* UNCLASSIFIED	21. NO. OF PAGES
	20. SECURITY CLASS *(This page)* UNCLASSIFIED	22. PRICE

INSTRUCTIONS

1. REPORT NUMBER
Insert the EPA report number as it appears on the cover of the publication.

2. LEAVE BLANK

3. RECIPIENTS ACCESSION NUMBER
Reserved for use by each report recipient.

4. TITLE AND SUBTITLE
Title should indicate clearly and briefly the subject coverage of the report, and be displayed prominently. Set subtitle, if used, in smaller type or otherwise subordinate it to main title. When a report is prepared in more than one volume, repeat the primary title, add volume number and include subtitle for the specific title.

5. REPORT DATE
Each report shall carry a date indicating at least month and year. Indicate the basis on which it was selected *(e.g., date of issue, date of approvcl, date of preparation, etc.).*

6. PERFORMING ORGANIZATION CODE
Leave blank.

7. AUTHOR(S)
Give name(s) in conventional order *(John R. Doe, J. Robert Doe, etc.).* List author's affiliation if it differs from the performing organization.

8. PERFORMING ORGANIZATION REPORT NUMBER
Insert if performing organization wishes to assign this number.

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Give name, street, city, state, and ZIP code. List no more than two levels of an organizational hirearchy.

10. PROGRAM ELEMENT NUMBER
Use the program element number under which the report was prepared. Subordinate numbers may be included in parentheses.

11. CONTRACT/GRANT NUMBER
Insert contract or grant number under which report was prepared.

12. SPONSORING AGENCY NAME AND ADDRESS
Include ZIP code.

13. TYPE OF REPORT AND PERIOD COVERED
Indicate interim final, etc., and if applicable, dates covered.

14. SPONSORING AGENCY CODE
Leave blank.

15. SUPPLEMENTARY NOTES
Enter information not included elsewhere but useful, such as: Prepared in cooperation with, Translation of, Presented at conference of, To be published in, Supersedes, Supplements, etc.

16. ABSTRACT
Include a brief *(200 words or less)* factual summary of the most significant information contained in the report. If the report contains a significant bibliography or literature survey, mention it here.

17. KEY WORDS AND DOCUMENT ANALYSIS
(a) DESCRIPTORS - Select from the Thesaurus of Engineering and Scientific Terms the proper authorized terms that identify the major concept of the research and are sufficiently specific and precise to be used as index entries for cataloging.

(b) IDENTIFIERS AND OPEN-ENDED TERMS - Use identifiers for project names, code names, equipment designators, etc. Use open-ended terms written in descriptor form for those subjects for which no descriptor exists.

(c) COSATI FIELD GROUP - Field and group assignments are to be taken from the 1965 COSATI Subject Category List. Since the majority of documents are multidisciplinary in nature, the Primary Field/Group assignment(s) will be specific discipline, area of human endeavor, or type of physical object. The application(s) will be cross-referenced with secondary Field/Group assignments that will follow the primary posting(s).

18. DISTRIBUTION STATEMENT
Denote releasability to the public or limitation for reasons other than security for example "Release Unlimited." Cite any availability to the public, with address and price.

19. & 20. SECURITY CLASSIFICATION
DO NOT submit classified reports to the National Technical Information service.

21. NUMBER OF PAGES
Insert the total number of pages, including this one and unnumbered pages, but exclude distribution list, if any.

22. PRICE
Insert the price set by the National Technical Information Service or the Government Printing Office, if known.