AD-751 542

A NEW FALLOUT TRANSPORT CODE FOR THE DELFIC SYSTEM: THE DIFFUSIVE TRANSPORT MODULE. SUPPLEMENT

Hillyer G. Norment

Mount Auburn Research Associates, Incorporated

Prepared for:

Defense Nuclear Agency

October 1972

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

AD751542

DNA 2669
Supplement
October 1972

A NEW FALLOUT TRANSPORT CODE FOR THE DELFIC SYSTEM: THE DIFFUSIVE TRANSPORT MODULE - SUPPLEMENT

Prepared by

H. G. Norment

MT. AUBURN RESEARCH ASSOCIATES, INC.
385 Elliot Street
Newton, Massachusetts 02164

This work was supported by the Defense Nuclear Agency
under NWED/Subtask PD068-04

Sponsored by

Defense Nuclear Agency
Washington, D. C. 20305

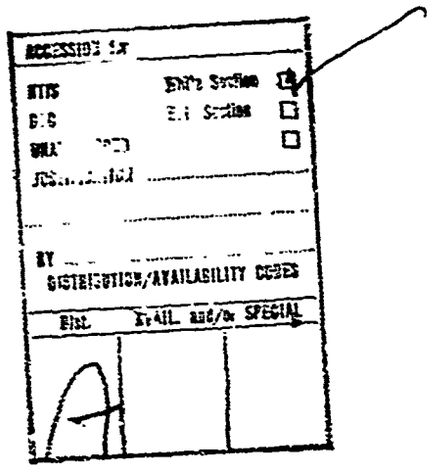
DDC
RECEIVED
NOV 22 1972
B

Contract No. DASA01-71-C-0118

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151

Approved for public release; distribution unlimited

Destroy this report when it is no longer needed. Do not return to sender.



Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)
MT. AUBURN RESEARCH ASSOCIATES, INC.
385 Elliot Street
Newton, Massachusetts 02164

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE
A New Fallout Transport Code for the DELFIC System: The Diffusive Transport Module - Supplement

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Final Report, 6 May 1971 - 6 May 1972. Approved 3 October 1972.

5. AUTHOR(S) (First name, middle initial, last name)
Hillyer G. Norment

6. REPORT DATE
October 1972

7a. TOTAL NO. OF PAGES
~~48~~ 43

7b. NO. OF REFS
0

8a. CONTRACT OR GRANT NO.
DASA01-71-C-0118

b. PROJECT NO.
NWED:XAXP

c. Task & Subtask: D068

d. Work Unit: 04

9a. ORIGINATOR'S REPORT NUMBER(S)

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
DASA 2669 Supplement

10. DISTRIBUTION STATEMENT
Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Defense Nuclear Agency
Washington, D. C. 20305

13. ABSTRACT

The Diffusive Transport Module of the new DELFIC fallout prediction system has undergone additional development since publication of its description in DASA 2669. This supplement to DASA 2669 describes these developments and presents amendments and corrections to the code and its documentation. Complete FORTRAN statement listings of subroutines that have been substantially changed are included.

DD FORM 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified

Unclassified
Security Classification

14 KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Nuclear weapons effects						
Radioactivity						
Fallout prediction						
DELFIC						

unclassified

DNA 2669
Supplement

October 1972

A NEW FALLOUT TRANSPORT CODE FOR THE DELFIC SYSTEM: THE DIFFUSIVE TRANSPORT MODULE - SUPPLEMENT

Prepared by

H. G. Norment

MT. AUBURN RESEARCH ASSOCIATES, INC.
385 Elliot Street
Newton, Massachusetts 02164

This work was supported by the Defense Nuclear Agency
under NWED/Subtask PD068-04

Sponsored by

Defense Nuclear Agency
Washington, D. C. 20305

Contract No. DASA01-71-C-0118

Approved for public release; distribution unlimited

ABSTRACT

The Diffusive Transport Module of the new DELFIC fall-out prediction system has undergone additional development since publication of its description in DASA 2669. This supplement to DASA 2669 describes these developments and presents ammendments and corrections to the code and its documentation. Complete FORTRAN statement listings of sub-routines that have been substantially changed are included.

TABLE OF CONTENTS

		Page
1.	INTRODUCTION	1
2.	MODEL REVISIONS	3
3.	DOCUMENTATION REVISIONS	5
4.	CODE REVISIONS	11

1. INTRODUCTION

Since publication of DASA 2669*, development and application of the DELFIC Diffusive Transport Module (DTM) has continued. A few revisions to the model have been made, several of them of major importance. In addition, some hidden "bugs" have been uncovered and corrected. In this supplement to DASA 2669 we describe the important model revisions, and amend the documentation. We also correct errors in the documentation, and provide FORTRAN statement listings of subroutines that have been changed substantially.

---

* H. G. Norment and E. J. Tichovolsky, "A New Fallout Transport Code for the DELFIC System: The Diffusive Transport Module," ARCON Corporation Report R71-1W, DASA 2669 (1 March 1971), AD 727 613.

2. MODEL REVISIONS

2.1 Initial Parcel Description

Fallout parcels are taken to be distributed in the horizontal about their centers of mass by a Gaussian density function. The initial Gaussian standard deviation was set equal to the parcel radius that was received via tape IPARIN from the Cloud Rise-Transport Interface Module. This has been changed so that the initial standard deviation is one-half the input parcel radius. (Compare subroutine SPRVS card 81 of DASA 2669 with SPRVS card 94 of this supplement.)

2.2 Simple Advection-Plus-Settling

In some cases, it is desirable to transport fallout in a simple advection-plus-settling mode; that is, without accounting for diffusion in the vertical. In this mode, integration of Eq. (16) is bypassed, and the parcel trajectory is computed via Eq. (32). As actually employed in the original DTM, Eq. (32) was modified to the form

$$\vec{r}_c = \vec{r}_i + \frac{1}{\langle f\rangle - \langle w\rangle} \sum_{z_i}^{z_g} \vec{U}(x,y,z,t)\Delta z \ ,$$

where the average settling speed, $\langle f\rangle$, was taken to be

$$\langle f\rangle = \frac{f(z_i) + f(z_g)}{2}$$

and $\langle w\rangle$ was an average vertical air velocity. For cases where $z_i - z_g$ is large, there can be significant differences in the particle settling speeds in the upper layers compared with those in the lower layers. When, in addition, there is large wind shear in the vertical,

Preceding page blank

it becomes necessary to use settling speeds that are computed individually for each wind layer instead of an average settling speed. The code has been changed so that this is done.

Major changes have been required in several programs. The most extensive changes are in subroutines SPRVS and TRANP. A new array, CAVS(KBHF), is created to store a table of particle settling speeds; it contains an entry for each wind layer.

Before a parcel is transported via this mode, a test is made to determine if the parcel can impact in the time allowed for transport. This test is simplified by using the knowledge that all parcels comprised of particles of a particular size class are processed sequentially in one group. Thus, as each new particle size class is encountered, the altitude above which these particles cannot impact is computed. Then, any parcel in the group whose base is above this altitude is bypassed. To perform the altitude limit calculation, an average vertical wind velocity is needed for each wind layer. To accommodate this, the array WAVG(LTIMF) was changed to the array WAVG(KBHF,LTIMF).

3. DOCUMENTATION REVISIONS

Revisions and corrections are intermixed and listed in order of their encounter in DASA 2669.

Page 27, Eq. (23):

$$t_1 = 4c_3 \left(\frac{\sigma_{t_i}^2}{\varepsilon} \right)^{1/3} \tag{23}$$

Page 27, line 16:

In our application, σ_{t_i} is taken to be one-half of the radius of a cloud wafer as recorded

Page 44, line 3:

where the summation is over the N data with the largest f_i, and the weighting factors,

Page 44, lines 6 and 7:

The parameters α, β, and N are specified by the user and the x_i and y_i are relative to the n-th lattice cell center. The calculations of the f_i are performed so that whenever a factor in Eq. (39) is found to

Page 44, lines 9 and 10:

than the total number of observations, M, only the N observations with the largest f_i relative to the n-th lattice cell center are considered in the calculations.

Page 45, lines 8-12:

list, two tables of settling rates for this particle are computed. One table contains an entry for each altitude increment used in the numerical integration of Eq. (16).

The other table contains an entry for each of the larger altitude increments that are used to resolve the wind and turbulence fields. These same tables are used through transport for this parcel and the remainder of the parcels in the first group. When the first parcel of the second group is selected, new tables are computed, and so on.

Page 49, Eq. (45):

$$\frac{G_j^{n+1} - G_j^n}{\Delta t} = \frac{1}{2(\Delta z)^2} \left[(K_{j+1} + K_j)G_{j+1}^n - (K_{j+1} + 2K_j + K_{j-1})G_j^n \right.$$

$$\left. + (K_j + K_{j-1})G_{j-1}^n\right] + \frac{1}{\Delta z}\left[(f_{j+1} - w_{j+1})G_{j+1}^n - (f_j - w_j)G_j^n\right] \quad (45)$$

Page 53, following Eq. (65):

where in the code K_{J+1} is taken to be equal to K_J.

Page 66, paragraphs 1 and 2:

Beginning at its input location and time, the parcel base or top is transported via local winds in the cell of its residence. At the same time it settles at a speed that is computed for the altitude at the center of the wind cell of its residence. When it passes through a wind cell boundary or a time boundary, the wind and settling speed are changed to those of the new cell or update. This continues until ground impaction occurs or until an extreme wind field or time boundary is encountered. The calculation requires one step for each cell through which the base or top passes.

When an extreme boundary is encountered by a top or base, the location and time of the encounter is recorded. These values are used in the definition of the deposit increment as described on page 70. The altitude of a deposit increment is always recorded as the arithmetic average of the impact

altitudes of its top and base. Thus, the recorded altitude of a deposit increment that has reached an extreme boundary can be well above the deposition plane.

Page 71, Figure 9:

The quantity labeled $\sigma(\perp)_{\perp}$ that lies to the left of the deposit increment ellipse should be replaced by $\sigma(\perp)_d$.

Page 88, lines 10 and 11:

DFZ. An area-weighted average vertical wind, WAVG(KBH, LTIM), is derived from array WFZ for each altitude layer and update. Likewise, a volume-weighted

Page 88, equation for DKAV:

$$DKAV = \sum_{KBH=1}^{KBHX-1} \frac{DFZ(KBH,NDATA,LTIM)*(ZBH(KBH+1) - ZBH(KBH))}{ZBH(KBHX) - ZBH(1)}$$

Page 88, lines 27 and 28:

out on ISOUT. In a parallel operation the quantities WAVG(KBH,LTIM), which are area-weighted vertical wind velocities for each wind layer and update, are computed and printed out on

Page 96, line 25:

in the horizontal RWFR(J)/2.0;

page 97, line 12:

ZPAR(J), PSAM(J), RWFR(J)/2.0, DWFR(J), ZLWF(J), and VWFR(J) are

Page 98, line 2:

for further details.) Also computed are the settling speeds,

CAVS(KBH), for each altitude, and the altitude, ZLIM, above which deposition is impossible, via gravity settling in the specified wind field, for particles in the particular size class being considered.

Page 100, last line:

CAV = CAV - WAVGK

Page 101, first line:

WAVGK and DAVG(1) are the average vertical components of wind

Page 101, line 15:

ground by advection at fall rates CAVS(KBHF) via a call to subroutine ADVEC.

Page 101, line 23:

limit: i.e., whenever ZLOW $\geq$ ZLIM. The comment

Page 110, lines 16-19:

also obtained via the COMMON area QPARM. The particle settling speeds, CAVS(KBHF), and area-weighted vertical air velocities, WAVG(KBHF,LTIMF), for each wind layer, are obtained

Page 110, line 27:

parcel base is advected, while settling at speeds CAVS(KBH) - WAVG(KBH,LTIM), from position (XP,YP,ZP =

Page 110, line 29:

TOL. The standard deviations of the

Page 110, line 33:

parcel top is advected, while settling at speeds CAVS(KBH) - WAVG(KBH,LTIM), from position (XP,YP,

Page 111, line 2:

at time TOU. The standard

Page 111, second paragraph:

After both base and top of a parcel have been transported, the arithmetic average of the two impact times and impact altitudes are recorded for the deposit increment (as shown on the next page). In the event that the base or top encounters an extreme wind field or time boundary during transport, subroutine TRANP returns the coordinates of the encounter point. Therefore, the altitude recorded for a deposit increment can be well above the deposition plane.

Page 111, third paragraph, lines 4 and 5:

is considered superfluous whenever ZP differs from ZDEP by less than 0.1. Instead, XOL, YOL, ZOL, TOL, SIGXL, SIGYL,

Page 122, line 28:

When vertical diffusive transport is employed, the computation of horizontal parcel advection is based on the

Page 123, insert between the second and third paragraphs:

For transport via simple advection plus settling, TRANP is called by subroutine ADVEC with KRIP=0 (otherwise KRIP=1). The parcel top or bottom is transported stepwise through the vertical layers between ZP and ZDEP via the layerwise mode (see below). In each layer, with index KBH, the vertical velocity is

WBAR = WFZ(KBH, NDATO, ATIM) - CAVS(KBH)

and KAY = - .1. TSEG is computed as shown by the equation in the preceding paragraph.

Page 123, third paragraph, line 3:

is traversed at a time. This mode is mandatory for transport via simple advection plus settling, or when a parcel trajectory

Page 125, line 6:

The rapid computation mode is employed for vertical diffusive transport when

Page 145, insert at the end of the Input Data Card 4 discussion:

To preempt vertical diffusive transport, set KX=1 and set ZMAX arbitrarily large. This causes all parcels to be transported via the simple-advection-plus-settling mode (Eq. (32)). Of course, horizontal diffusive growth of parcels is accounted for in any case.

Page 146, line 14:

CSKIP - 0.1

Page 152, line 8:

HITIME. If the vertical diffusive mode of transport is used, the KBHX'th base should be above or at the top of the transport space as this top is specified by ZMAX.

Page 156, Table 4, Record Number 12, line 6 under Content:

size class central diameter (μm), mass of fallout (kg)

Page 154, to the end of paragraph 1 add:

However, the turbulent energy dissipation rates can be input only for the horizontal directions; for the vertical direction Fickian diffusivities always are input, regardless of which type of data are input for the horizontal.

4. CODE REVISIONS

4.1 Single Card Changes

Addition of the arrays CAVS(KBHF) and WAVG(KBHF,LTIMF) requires revisions in DIMENSION statements and subroutine argument lists. However, complete FORTRAN statement listings are given in this supplement for all subroutines that require these revisions.

Subroutine BOUN, card 27:

```
CALL NEST (NET, NETSU, XO, YO, NDATO, XL, XR, YL, YU, ICF,
JCF, NCF)
```

Subroutine DUMPER:

Place card 33 in its proper position.

Subroutine NEST, insert between cards 20 and 21:

```
DIMENSION NET(ICF, JCF), NETSU(NCF)
```

4.2 FORTRAN Statement Listings

Complete FORTRAN statement listings are given for the following subroutines. These subroutines are operational on the UNIVAC 1108.

Subroutine	Page
C31M	13
ADMIN	18
ADVEC	21
AMBNT	23
SPRVS	26
TRANP	32

The machine used to prepare these listings prints a # symbol to represent a 4-8 punch; this symbol should be an apostrophe ('). In FORMAT and DATA statements, the apostrophe is used to define Hollerith character fields.

```
C       SEPTEMBER 1971                                                  C31M   1
C       C31M IS THE MAIN PROGRAM WHICH DIRECTS THE DIFFUSIVE TRANSPORT  C31M   2
C       MODULE OPERATIONS.THE OBJECT-TIME DIMENSIONS ARE SET IN C31M.   C31M   3
C       THESE DIMENSIONS AND THEIR RESPECTIVE ARRAYS ARE                C31M   4
C       KKF    - AA,BB,CC,DENOM,E,F,Q,R,DIFF,ZHT                        C31M   5
C       LTIMF  - TIMUP,DAVG,WAVG                                        C31M   6
C       KBHF   - ZBH,ZCH                                                C31M   7
C       NDATF  - KTOPO                                                  C31M   8
C       KBHF, NDATF, LTIMF - DFZ,DXSUM,DYSUM,USUM,VSUM,WFZ,RSUM         C31M   9
C       ICF, JCF - NET                                                  C31M  10
C       NCF    - NETSU                                                  C31M  11
C       MARF   - MARY                                                   C31M  12
C       NATF   - ALT,ATEMP,RHO                                          C31M  13
C       DATA LITERALS MUST BE INSERTED IN THE DIMENSION STATEMENTS AND IN C31M  14
C       THE RIGHT HAND SIDES OF THE ARITHMETIC STATEMENTS IN WHICH THE  C31M  15
C       ABOVE VARIABLE NAMES APPEAR.                                    C31M  16
C       ***************************** GLOSSARY ******************************C31M  17
C       AA(K) - EQUALS S2*(DIFF(K+1)+DIFF(K))+S1*F(K+1)                 C31M  18
C       ALT   - ALTITUDES FOR ATMOS. DENSITY AND VISCOSITY TABLE        C31M  19
C       ATEMP - DYNAMIC VISCOSITY OF AIR DATA VECTOR FOR ATMOS. TABLE   C31M  20
C       BB(K) - EQUALS S2*(DIFF(K+1)+2.*DIFF(K)+DIFF(K-1))+S1*F(K)      C31M  21
C       CAV   - AVG. FALLRATE USED IN COMPUTING PDEST. IT APPLIES MID WAY C31M  22
C               FROM PARCEL TOP TO ZMIN.                                C31M  23
C       CAVS  - PARTICLE FALL RATE FOR EACH ATMOS. STRATUM              C31M  24
C       CC(K) - EQUALS S1*(DIFF(K)+DIFF(K-1))                           C31M  25
C       CROSS - CROSSWIND CROSSING TRAJECTORIES CORRECTION TO TURB.     C31M  26
C       CSKIP - TOTAL FRACTIONAL PARCEL DEPOSITION THRESHOLD            C31M  27
C       DAVG  - AVG. ATMOS. VERT. TURB. PER UPDATE DATA VECTOR          C31M  28
C       DENOM(K) - EQUALS 1.-THETA*(BB(K)-CC(K)*E(K-1))                 C31M  29
C       DEP   - DEPOSITED FRACTIONAL MASS INCREMENT                     C31M  30
C       DIFF(K) - VERT. DIFFUSIVITY AT K-TH SMALL ALTITUDE INCREMENT    C31M  31
C       DFKXS1- VERTICAL DFFUSIVITY AT ALTITUDE INCREMENT KX-1          C31M  32
C       DFZ   - TURBULENCE Z COMPONENT 3-DIM. DATA ARRAY                C31M  33
C       DINCR - RATE OF CHANGE OF FRACTIONAL MASS DEPOS. RATE THRESH.   C31M  34
C       DOPEN - MASS DEPOSITION RATE THRESHOLD                          C31M  35
C       DOWN  - DOWNWIND CROSSING TRAJECTORIES CORRECTION TO TURB.      C31M  36
C       DT    - SMALL ITERATION TIME STEP FOR VERT. DIFF. DIFF. EQ.     C31M  37
C       DWAF  - PARCEL VERT. THICKNESS BEFORE ADVECTION                 C31M  38
C       DXSUM - TURBULENCE X COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY C31M  39
C       DYSUM - TURBULENCE Y COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY C31M  40
C       DZ    - SMALL ALTITUDE INCREMENT FOR VERT. DIFF. DIFF. EQ.      C31M  41
C       DZMIN - MINIMUM VALUE OF DZ                                     C31M  42
C       E(K)  - EQUALS THETA*AA(K)/DENOM(K)                             C31M  43
C       EDDY  - RATIO OF LAGRANGIAN TURBULENCE TIME SCALE TO EULERIAN   C31M  44
C               TURBULENCE LENGTH SCALE                                 C31M  45
C       EFFLUX  UPPER EFFLUX FRACTIONAL MASS                            C31M  46
C       F(K)  - IN SUB. DIFFE, WORKING SPACE IMPLICIT METHOD DATA VECTOR. C31M  47
C               IN SUB. AMBNT, WORKING SPACE FOR VERTICAL VELOCITIES    C31M  48
C       FAV   - MID-ATMOS. AVG. FALLRATE. USED IN CROSSING TRAJECTORIES C31M  49
C               CORRECTIONS AND IN TRUNCATION ERROR ESTIMATION .        C31M  50
C       FMAB  - CUMULATIVE FRACTIONAL MASS AIRBORNE                     C31M  51
C       FMBEL - MIN. PARCEL FRACTIONAL MASS ALOFT TO BE TRANSPORTED     C31M  52
C       ICF   - MAX. FORMAL DIM. CORRESPONDING TO ICX                   C31M  53
C       ICX   - OBJECT-TIME FIRST MAX. DIM. OF ARRAY NET. NUMBER OF NET C31M  54
C               MESHES IN EAST-WEST ROW.                                C31M  55
C       IPARIN- LOGICAL UNIT NUMBER OF  CR-TRANS. INTER. MOD. OUTPUT TAPE C31M  56
C       IPOUT - LOGICAL UNIT NUMBER OF DIFF. TRANS. MOD. OUTPUT TAPE    C31M  57
C       ISIN  - LOGICAL UNIT NUMBER OF SYSTEM INPUT TAPE                C31M  58
```

```
C     ISOUT - LOGICAL UNIT NUMBER OF SYSTEM OUTPUT TAPE                C31M  59
C     JCF   - MAX. FORMAL DIM. CORRESPONDING TO JCX                    C31M  60
C     JCX   - OBJECT-TIME SECOND MAX. DIM. OF ARRAY NET. NUMBER OF NET C31M  61
C             MESHES IN SOUTH-NORTH ROW.                               C31M  62
C     KBH   - ATMOS. VERT.  SPACE INDEX FOR ARRAYS USUM,VSUM,WFZ,DXSUM, C31M  63
C             DYSUM, DFZ, RSUM, ZBH, ZCH                               C31M  64
C     KBHF  - MAX. FORMAL DIM. CORRESPONDING TO KBHX                   C31M  65
C     KBHX  - OBJECT-TIME MAX. VALUE OF KBH                            C31M  66
C     KKF   - MAX. FORMAL DIM. CORRESPONDING TO KKX                    C31M  67
C     KKM   - ALWAYS EQUALS 2. CORRESPONDS TO K=0 ALTITUDE INCREMENT   C31M  68
C     KKMA1 - EQUALS KKM+1                                             C31M  69
C     KKX   - EQUALS KX+KKM                                            C31M  70
C     KKXS1 - EQUALS KKX-1                                             C31M  71
C     KPIP   - CONTROL VARIABLE                                        C31M  72
C                0 FOR ADVECTIVE TRANSPORT                             C31M  73
C                1 FOR DIFFUSIVE TRANSPORT                             C31M  74
C     KTOPO - NET MESH AND SUB-MESH TOPOGRAPHY TABLE DATA VECTOR       C31M  75
C     KX    - MAX. NUMBER OF DZ ALTITUDE INCREMENTS                    C31M  76
C     KXMIN - MIN. NUMBER OF DZ ALTITUDE INCREMENTS                    C31M  77
C     LSTEP - NUMBER OF IMPLICIT METHOD ITERATIONS. TDELT=LSTEP*DT.    C31M  78
C             DYSUM, DFZ, RSUM, TIMUP, DAVG, HAVG                      C31M  79
C     LTIM  - ATMOS. UPDATE TIME  INDEX FOR ARRAYS USUM,VSUM,WFZ,DXSUM, C31M  80
C     LTIMX - OBJECT-TIME MAX. VALUE OF LTIM                           C31M  81
C     LTIMF - MAX. FORMAL DIM. CORRESPONDING TO LTIMX                  C31M  82
C     MARF  - MAX. FORMAL DIM. CORRESPONDING TO MARX                   C31M  83
C     MARX  - OBJECT-TIME MAX. DIM. OF ARRAY MARY                      C31M  84
C     MARY  - HORIZ. ATMOS. SPACE RESOLUTION NET MESH AND SUB-MESH     C31M  85
C             CONTROL FLAGS DATA VECTOR                                C31M  86
C                                                                      C31M  87
C     MC    - CONTROL INTEGER DATA VECTOR                              C31M  88
C                                                                      C31M  89
C             MC(1) LESS THAN OR EQUAL TO ZERO, SUPPRESSES LISTING OF  C31M  90
C             EXPANDED WIND AND TURB. DATA                             C31M  91
C                                                                      C31M  92
C             MC(1) GREATER THAN OR EQUAL TO ONE, CAUSES LISTING OF    C31M  93
C             WIND AND TURB. DATA BEFORE SUMMATION                     C31M  94
C                                                                      C31M  95
C             MC(1) EQUALS TWO, CAUSES LISTING OF WIND AND TURB. DATA  C31M  96
C             AFTER SUMMATION                                          C31M  97
C                                                                      C31M  98
C             MC(2) EQUALS ONE, SUPPRESSES LISTING OF ATMOS. VISC. AND C31M  99
C             DENS. TABLES                                             C31M 100
C                                                                      C31M 101
C             MC(3) EQUALS ZERO, SUPPRESSES LISTING OF DEPOSIT         C31M 102
C             INCREMENTS ON TAPE ISOUT                                 C31M 103
C                                                                      C31M 104
C             MC(4) EQUALS ONE, CAUSES PRINTOUT OF TRANSPORT           C31M 105
C             INTERMEDIATE RESULTS ON TAPE ISOUT. WARNING. PRINTOUT    C31M 106
C             IS EXTRAORDINARILY VOLUMINOUS. FOR DEBUGGING ONLY.       C31M 107
C                                                                      C31M 108
C             MC(7) EQUALS ONE, SUPPRESSES LISTING OF RAW WIND AND TURB.C31M 109
C             INPUT DATA                                               C31M 110
C                                                                      C31M 111
C             MC(10) EQUALS ONE, CAUSES TURB. DATA TO BE TREATED AS    C31M 112
C             KOLMOGOROFF-BATCHELOR ENERGY DISSIPATION RATES           C31M 113
C                                                                      C31M 114
C             MC(10) NOT EQUAL TO ONE, CAUSES TURB. DATA TO BE TREATED C31M 115
C             AS FICKIAN DIFFUSIVITIES                                 C31M 116
```

```
C                                                                       C31M 117
C             MC(18) EQUALS ONE, SUPPRESSES READING FROM TAPE IPARIN    C31M 118
C             AND WRITING ONTO TAPE IPOUT                               C31M 119
C                                                                       C31M 120
C      MINT   - MIN. NUMBER OF DT SMALL TIME STEPS PER DEPOSIT INCREMENT C31M 121
C             TIME INTERVAL                                             C31M 122
C      NAT    - NUMBER OF ALTITUDE STRATA IN ATMOS. DENS. AND VISC. TABLE C31M 123
C      NATF   - MAX. FORMAL DIM. CORRESPONDING TO NATF                  C31M 124
C      NBLK   - RECORD BLOCK SIZE FOR DEPOSIT INCREMENT RECORDS         C31M 125
C      NCF    - MAX. FORMAL DIM. CORRESPONDING TO NCX                   C31M 126
C      NCX    - OBJECT-TIME MAX. DIM. OF ARRAY NETSU                    C31M 127
C      NDATA  - ATMOS. HORIZ. SPACE INDEX FOR ARRAYS USUM,VSUM,WFZ,DXSUM, C31M 128
C             DYSUM, DFZ, RSUM, KTOPO                                   C31M 129
C      NDATC  - HORIZONTAL INDEX OF LATTICE CELL CONTAINING POINT (XC,YC) C31M 130
C      NDATO  - HORIZONTAL INDEX OF LATTICE CELL CONTAINING POINT (XO,YO) C31M 131
C      NDATP  - HORIZONTAL INDEX OF LATTICE CELL CONTAINING POINT (XP,YP) C31M 132
C      NDATX  - OBJECT-TIME MAX. VALUE OF NDATA                          C31M 133
C      NDATF  - MAX. FORMAL DIM. CORRESPONDING TO NDATX                  C31M 134
C      NDELT  - NOMINAL NUMBER OF DEPOSIT INCREMENTS PER FALLOUT PARCEL  C31M 135
C      NET    - HORIZONTAL SPACE CONTROL NET MESH 2-DIM. ARRAY           C31M 136
C      NETSU  - HORIZONTAL SPACE CONTROL NET SUB-MESH DATA VECTOR        C31M 137
C      NSEQC  - STORAGE SEQUENCE ORDINAL OF FIRST PARCEL TO BE TRANSPORTEDC31M 138
C      PHI    - EQUALS 1-THETA                                           C31M 139
C      Q(K)   - CONCENTRATION IN K-TH ALTITUDE INCREMENT                 C31M 140
C      RHO    - ATMOS. DENSITY DATA VECTOR FOR ATMOS. TABLE              C31M 141
C      RO     - WIND HEADING ORIENTATION ANGLE AFTER ADVECTION           C31M 142
C      ROPART- FALLOUT PARTICLE DENSITY                                  C31M 143
C      RSUM   - WIND HEADING ORIENTATION ANGLE (WEIGHTED SUM) 3-DIM. ARRAYC31M 144
C      RWAF   - PARCEL RADIUS IN PARCEL CENTRAL PLANE BEFORE ADVECTION   C31M 145
C      SIGXO  - PARCEL MASS HOR. STAND. DEV. DOWNWIND AFTER ADVECTION    C31M 146
C      SIGYO  - PARCEL MASS HOR. STAND. DEV. CROSSWIND AFTER ADVECTION   C31M 147
C      S1     - EQUALS DT/DZ                                             C31M 148
C      S2     - EQUALS DT/(2.*(DZ)**2))                                  C31M 149
C      TDELT  - CURRENT DEPOSIT INCREMENT TIME INTERVAL                  C31M 150
C      TDEP   - ADVECTIVE TRANSPORT TIME INTERVAL                        C31M 151
C      THETA  - IMPLICIT FINITE DIFFERENCE PARAMETER                     C31M 152
C      THETO  - DOUBLE PRECISION WORD CORRESPONDING TO THETA             C31M 153
C      TIME   - TIME AT ONSET OF CURRENT DEPOSIT INCREMENT TIME INTERVAL C31M 154
C      TIMEX  - SIMULATED TRANSPORT TIME LIMIT                           C31M 155
C      TIMUP  - ATMOSPHERE UPDATE TIMETABLE DATA VECTOR                  C31M 156
C      TO     - TIME AFTER PARCEL ADVECTION                              C31M 157
C      TP     - TIME BEFORE PARCEL ADVECTION                             C31M 158
C      TPAUS  - TIME AT END OF CURRENT DEPOSIT INCREMENT TIME INTERVAL   C31M 159
C      USUM   - WIND X COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY        C31M 160
C      VETA   - VERTICAL DIFFUSION ABSORPTION COEFFICIENT                C31M 161
C      VSUM   - WIND Y COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY        C31M 162
C      W(K)   - SETTLING RATE AT K-TH SMALL ALTITUDE INCREMENT           C31M 163
C      WAVG   - AVG. ATMOS. VERT. WIND  PER UPDATE PER STRATUM           C31M 164
C      WAVGK  - WAVG AVERAGED OVER THE STRATA FOR THE FIRST UPDATE       C31M 165
C      WFZ    - WIND Z COMPONENT 3-DIM. DATA ARRAY                       C31M 166
C      WINT   - NET CONTROL MESH DIMENSION                               C31M 167
C      XLLC   - X COORDINATE OF SOUTH-WEST CORNER OF ATMOS. SPACE        C31M 168
C      XO     - PARCEL CENTER X COORDINATE AFTER ADVECTION               C31M 169
C      XP     - PARCEL CENTER X COORDINATE BEFORE ADVECTION              C31M 170
C      YLLC   - X YOORDINATE OF SOUTH-WEST CORNER OF ATMOS. SPACE        C31M 171
C      YO     - PARCEL CENTER Y COORDINATE AFTER ADVECTION               C31M 172
C      YP     - PARCEL CENTER Y COORDINATE BEFORE ADVECTION              C31M 173
C      ZBH    - ATMOSPHERE STRATA BASE-ALTITUDE DATA VECTOR              C31M 174
```

```
C     ZCH   - ATMOSPHERE STRATA MID-ALTITUDE DATA VECTOR                C31M 175
C     ZDEP  - ADVECTIVE TRANSPORT TERMINAL ALTITUDE                     C31M 176
C     ZHT(K) - K-TH ALTITUDE INCREMENT ABOVE ZMIN LEVEL                 C31M 177
C     ZLOW  - PARCEL BASE ALTITUDE BEFORE ADVECTION                     C31M 178
C     ZMAX  - ATMOSPHERE TOP ALTITUDE (FOR VERT. DIFF. DIFF. EQ.)       C31M 179
C     ZMIN  - GROUND LEVEL OR DEPOSITION PLANE ALTITUDE                 C31M 180
C     ZO    - PARCEL CENTER Z COORDINATE AFTER ADVECTION                C31M 181
C     ZP    - PARCEL CENTER Z COORDINATE BEFORE ADVECTION, EXCEPT AS    C31M 182
C             REDEFINED IN SUB. ADVEC                                   C31M 183
C     ZUPP  - PARCEL TOP ALTITUDE BEFORE ADVECTION. ZLOW+DWAF           C31M 184
      DOUBLE PRECISION AA(204),BB(204),CC(204),DENOM(204),E(204),F(204) C31M 185
      DOUBLE PRECISION Q(204)                                           C31M 186
      DIMENSION ZBH(    27),ZCH(    27),TIMUP(    6),MARY(    1)      C31M 187
      DIMENSION KTOPO(    1),NETSU(    1),NET(    1,    1)             C31M 188
      DIMENSION   DFZ(    27,    1,    6),  WFZ(    27,    1,    6)    C31M 189
      DIMENSION  USUM(    27,    1,    6), VSUM(    27,    1,    6)    C31M 190
      DIMENSION DXSUM(    27,    1,    6),DYSUM(    27,    1,    6)    C31M 191
      DIMENSION  RSUM(    27,    1,    6),DAVG(    6)                  C31M 192
      DIMENSION  WAVG(    27,    6)                                    C31M 193
      DIMENSION ALT(   260),RHO(   260),ATEMP(   260)                  C31M 194
      DIMENSION CAVS(   27),W(   204),DIFF(   204),ZHT(   204)         C31M 195
      COMMON /QPARM/ IPOUT ,IPARIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM     C31M 196
     1,NSEQO ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT   C31M 197
     2,EDDY  ,FMREL ,LSTEP ,MC(13),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN    C31M 198
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP      C31M 199
     4,DINCR ,DOWN  ,TP    ,ZLOW  ,DWAF  ,PWAF  ,ROPART,ZUPP  ,VETA    C31M 200
     5,DOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP          C31M 201
      ICF=1                                                             C31M 202
      ISIN=5                                                            C31M 203
      ISOUT=6                                                           C31M 204
      IPARIN=9                                                          C31M 205
      IPOUT=10                                                          C31M 206
      JCF=1                                                             C31M 207
      KBHF=27                                                           C31M 208
      KKF=204                                                           C31M 209
      LTIMF=6                                                           C31M 210
      MARF=1                                                            C31M 211
      NATF=260                                                          C31M 212
      NCF=1                                                             C31M 213
      NDATF=1                                                           C31M 214
      DO 1 N=1,NCF                                                      C31M 215
    1 NETSU(N)=0                                                        C31M 216
      DO 2 J=1,JCF                                                      C31M 217
      DO 2 I=1,ICF                                                      C31M 218
    2 NET(I,J)=0                                                        C31M 219
      DO 3 M=1,MARF                                                     C31M 220
    3 MARY(M)=0                                                         C31M 221
      DO 4 K=1,KBHF                                                     C31M 222
      ZBH(K)=0.                                                         C31M 223
    4 ZCH(K)=0.                                                         C31M 224
      DO 104 K=1,KBHF                                                   C31M 225
      DO 104 L=1,LTIMF                                                  C31M 226
  104 WAVG(K,L)=0.0                                                     C31M 227
      DO 5 N=1,NDATF                                                    C31M 228
    5 KTOPO(N)=0                                                        C31M 229
      DO 6 L=1,LTIMF                                                    C31M 230
      TIMUP(L)=0.                                                       C31M 231
      DAVG(L)=0.                                                        C31M 232
```

```
      DO 6 N=1,NDATF                                                    C31M 233
      DO 6 K=1,KBHF                                                     C31M 234
      DFZ(K,N,L)=0.                                                     C31M 235
      WFZ(K,N,L)=0.                                                     C31M 236
      USUM(K,N,L)=0.                                                    C31M 237
      VSUM(K,N,L)=0.                                                    C31M 238
      DXSUM(K,N,L)=0.                                                   C31M 239
      DYSUM(K,N,L)=0.                                                   C31M 240
    6 RSUM(K,N,L)=0.                                                    C31M 241
COMMENCE READING DATA INPUTS FROM TAPES ISIN AND IPARIN                 C31M 242
COMMENCE WRITING DATA OUTPUT HEADERS ONTO TAPES ISOUT AND IPOUT         C31M 243
      CALL        LINK(ALT,RHO,ATEMP,NATF)                              C31M 244
CONSTRUCT THE HORIZONTAL SPACE CONTROL NET                              C31M 245
      CALL        GETUP(NET,NETSU,KTOPO,HARY,MARF,ICF,JCF,NCF,NDATF)    C31M 246
CONSTRUCT THE ATMOSPHERIC STRATA AND UPDATE DATA VECTORS                C31M 247
      CALL        HITIMF(ZCH,ZBH,TIMUP,KBHF,LTIMF)                      C31M 248
CONTSRUCT AND FILL IN THE ATMOSPHERIC LATTICE AND UPDATE STRUCTURE      C31M 249
      CALL        ADMIN(NET,NETSU,ZBH,ZCH,TIMUP,USUM,VSUM,DXSUM,DYSUM,  C31M 250
     1RSUM,DFZ,WFZ,DAVG,WAVG,ICF,JCF,NCF,KBHF,NDATF,LTIMF)              C31M 251
CIRCUMVENT ALL TAPE HANDLING IF MC(18) EQUALS 1                         C31M 252
      IF(MC(18).EQ.1) GO TO 7                                           C31M 253
CALCULATE THE DIFFUSIVE TRANSPORT OF PARCELS ACCEPTED FROM TAPE IPARIN  C31M 254
COPY OUT RESULTS ONTO TAPE IPOUT                                        C31M 255
      CALL        SPRVS(NET,NETSU,ZBH,ZCH,TIMUP,USUM,VSUM,DXSUM,DYSUM,  C31M 256
     1RSUM,DFZ,WFZ,DAVG,WAVG,ALT,RHO,ATEMP,AA,BB,CC,DENOM,DIFF,E,F,Q,W, C31M 257
     2ZHT,ICF,JCF,NCF,KBHF,NDATF,LTIMF,KKF,NATF,CAVS)                   C31M 258
    7 CALL EXIT                                                         C31M 259
      STOP                                                              C31M 260
      END                                                               C31M 261
```

```
      SUBROUTINE ADMIN(NET,NETSU,ZBH,ZCH,TIMUP,USUM,VSUM,DXSUM,DYSUM,    ADMIN  1
     1RSUM,DFZ,WFZ,DAVG,WAVG,ICF,JCF,NCF,KBHF,NDATF,LTIMF)               ADMIN  2
C     SEPTEMBER 1971                                                    ADMIN  3
C                                                                       ADMIN  4
C     SUBROUTINE ADMIN CONSTRUCTS WIND DATA ARRAYS                      ADMIN  5
C                USUM, VSUM, WFZ, WAVG, RSUM                            ADMIN  6
C     AND TURBULENCE DATA ARRAYS                                        ADMIN  7
C                DXSUM, DYSUM, DFZ, DAVG.                               ADMIN  8
C     IN ADMIN ONLY LTIM AND SPEC ARE READ FROM TAPE ISIN.              ADMIN  9
C     SPEC  - DATA SPECIES IDENTIFICATION WORD ≠WIND≠ OR ≠SPEC≠         ADMIN 10
C     LTIM  - UPDATE ORDINAL OF DATA SET. FIRST ATMOS. SET  HAS LTIM=1. ADMIN 11
C     SUB. MKDAT IS CALLED TO PERFORM DATA EXTRAPOLATIONS.              ADMIN 12
C     AREA  - AREA OF HORIZ. SPACE NET                                  ADMIN 13
C     AREAN - AREA OF N-TH NET MESH OR SUB-MESH                         ADMIN 14
      COMMON /OPARM/ IPOUT ,IPARIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM      ADMIN 15
     1,NSEQO ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT    ADMIN 16
     2,FDDY  ,FMREL ,LSTEP ,MC(18),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN     ADMIN 17
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP      ADMIN 18
     4,DINCR ,DOWN  ,TP    ,ZLOW  ,DWAF  ,RWAF  ,ROPART,ZUPP  ,VETA    ADMIN 19
     5,DOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP           ADMIN 20
      DIMENSION RSUM(KBHF,NDATF,LTIMF),DAVG(LTIMF),WAVG(KBHF,LTIMF)     ADMIN 21
      DIMENSION NET(ICF,JCF),NETSU(NCF),ZCH(KBHF),TIMUP(LTIMF),ZBH(KBHF)ADMIN 22
      DIMENSION USUM(KBHF,NDATF,LTIMF),VSUM(KBHF,NDATF,LTIMF)           ADMIN 23
      DIMENSION DXSUM(KBHF,NDATF,LTIMF),DYSUM(KBHF,NDATF,LTIMF)         ADMIN 24
      DIMENSION DFZ(KBHF,NDATF,LTIMF),WFZ(KBHF,NDATF,LTIMF)             ADMIN 25
      DIMENSION LW(10),LD(10)                                           ADMIN 26
      INTEGER WIND,DFSN,DONE,SPEC                                       ADMIN 27
      DATA PROGRM/≠ADMIN ≠/                                             ADMIN 28
      DATA WIND/≠WIND≠/                                                 ADMIN 29
      DATA DFSN/≠DIFF≠/                                                 ADMIN 30
      DATA DONE/≠NO M≠/                                                 ADMIN 31
    1 FORMAT(≠0≠36X,≠UPDATE TIME INDEX≠I5,≠. WIND GRID CELL INDEX≠I5/)  ADMIN 32
    2 FORMAT(≠  WIND≠2(6X,≠HORIZONTAL≠),6X,≠ VERTICAL ≠6X,≠CROSSWIND ≠6XADMIN 33
     1,≠ DOWNWIND ≠6X,≠ VERTICAL ≠6X,≠HORIZONTAL≠)                      ADMIN 34
    3 FORMAT(≠ LAYER≠6X,≠E.-W. WIND≠6X,≠N.-S. WIND≠6X,≠   WIND   ≠3(6X,≠ADMIN 35
     2DIFFUSION ≠),6X,≠ ROTATION≠)                                      ADMIN 36
    4 FORMAT(≠ INDEX≠3(6X,≠ VELOCITY ≠),3(6X,≠ CONSTANT ≠),6X,≠  ANGLE≠)ADMIN 37
    5 FORMAT(≠ ≠I5,7E16.4)                                              ADMIN 38
    6 FORMAT(/25X,≠WEIGHTED SUMS OVER ABOVE COLUMN ENTRIES≠/)           ADMIN 39
    7 FORMAT(≠ ≠I5,2E16.4,16X,2E16.4,16X,E16.4)                         ADMIN 40
    8 FORMAT(≠ ≠22X,≠UPDATE≠I4,≠* CELL≠I4,                        ≠* ADMIN 41
     3AVG. VERT. DIFF. =≠E12.4)                                         ADMIN 42
    9 FORMAT(36X,I2,8X,3A4)                                             ADMIN 43
   10 FORMAT(/25X,≠ATMOSPHERE UPDATE≠I4,≠ FOR TIMES LATER THAN ≠E12.4,≠ ADMIN 44
     1SECONDS≠)                                                         ADMIN 45
   11 FORMAT(≠ ≠25X,≠* * * * * * * * * * WINDFIELD EXTRAPOLATION * * * *ADMIN 46
     1 * * * * *≠/)                                                     ADMIN 47
   12 FORMAT(≠ ≠25X,≠* * * * * * * * * DIFFUSIVITY EXTRAPOLATION * * * *ADMIN 48
     2 * * * * *≠/)                                                     ADMIN 49
   13 FORMAT(/25X,≠UPDATE≠I4,≠ OF THE WINDFIELD IS MISSING≠)            ADMIN 50
   14 FORMAT(/25X,≠UPDATE≠I4,≠ OF THE DIFFUSIVITY IS MISSING≠)          ADMIN 51
   15 FORMAT(≠ OVER ENTIRE HORIZONTAL GRID FOR UPDATE≠I4,               ADMIN 52
     6            ≠* AVG. VERT. DIFF. =≠E12.4)                          ADMIN 53
   16 FORMAT(≠  AVG. VERT. VEL. FOR EACH STRATUM IS - ≠)                ADMIN 54
   17 FORMAT(2X,I5,E16.4)                                               ADMIN 55
      AREA=ICX*JCX*(WINT**2)                                            ADMIN 56
      DO 999 L=1,LTIMX                                                  ADMIN 57
      LW(L)=L                                                           ADMIN 58
```

```
  999 LD(L)=L                                                           ADMIN 59
COPY IN LTIM AND SPEC FROM TAPE ISIN AND CALL SUB. MKDAT                ADMIN 60
 1000 READ(ISIN,9) LTIM,SPEC                                            ADMIN 61
      IF(SPEC.EQ.DONE) GO TO 1301                                       ADMIN 62
      IF((LTIM.LT.1).OR.(LTIM.GT.LTIMX)) CALL ERROR(PROGRM,-1000,ISOUT) ADMIN 63
      WRITE(ISOUT,10) LTIM,TIMUP(LTIM)                                  ADMIN 64
      IF(SPEC.EQ.WIND) GO TO 1101                                       ADMIN 65
      IF(SPEC.EQ.DFSN) GO TO 1201                                       ADMIN 66
      CALL ERROR(PROGRM,-1101,ISOUT)                                    ADMIN 67
 1101 WRITE(ISOUT,11)                                                   ADMIN 68
      DO 1102 L=1,LTIMX                                                 ADMIN 69
 1102 IF(LTIM.EQ.LW(L)) GO TO 1103                                      ADMIN 70
      CALL ERROR(PROGRM,-1102,ISOUT)                                    ADMIN 71
 1103 LW(L)=-1                                                          ADMIN 72
      CALL        MKDAT(ZCH,NET,NETSU,LTIM, USUM, VSUM,WFZ,ICF,JCF,NCF, ADMIN 73
     1KBHF,NDATF,LTIMF)                                                 ADMIN 74
      GO TO 1000                                                        ADMIN 75
 1201 WRITE(ISOUT,12)                                                   ADMIN 76
      DO 1202 L=1,LTIMX                                                 ADMIN 77
 1202 IF(LTIM.EQ.LD(L)) GO TO 1203                                      ADMIN 78
      CALL ERROR(PROGRM,-1202,ISOUT)                                    ADMIN 79
 1203 LD(L)=-1                                                          ADMIN 80
      CALL        MKDAT(ZCH,NET,NETSU,LTIM,DXSUM,DYSUM,DFZ,ICF,JCF,NCF, ADMIN 81
     1KBHF,NDATF,LTIMF)                                                 ADMIN 82
      GO TO 1000                                                        ADMIN 83
CHECK IF ANY WIND DATA SETS ARE MISSING                                 ADMIN 84
 1301 DO 1303 L=1,LTIMX                                                 ADMIN 85
      IF(LW(L).EQ.-1) GO TO 1302                                        ADMIN 86
      WRITE(ISOUT,13) LTIM                                              ADMIN 87
      CALL ERROR(PROGRM,-1302,ISOUT)                                    ADMIN 88
CHECK IF ANY TURBULENCE DATA SETS ARE MISSING                           ADMIN 89
 1302 IF(LD(L).EQ.-1) GO TO 1303                                        ADMIN 90
      WRITE(ISOUT,14) LTIM                                              ADMIN 91
      CALL ERROR(PROGRM,-1303,ISOUT)                                    ADMIN 92
 1303 CONTINUE                                                          ADMIN 93
CALCULATE THE WEIGHTED SUMS OVER ATMOS. STRATA AND REWRITE ARRAYS       ADMIN 94
C     USUM, VSUM, RSUM, DXSYM, DYSUM. ALSO COMPUTE DAVG AND WAVG.       ADMIN 95
      ZSPAN=ZBH(KBHX)-ZBH(1)                                            ADMIN 96
      DO 922 L=1,LTIMX                                                  ADMIN 97
      DO 1304 LK=1,KBHX                                                 ADMIN 98
 1304 WAVG(LK,L)=0.0                                                    ADMIN 99
      DAVG(L)=0.                                                        ADMIN100
      DO 921 N=1,NDATX                                                  ADMIN101
      DKAV=0.                                                           ADMIN102
      IF(MC(1).LT.1) GO TO 915                                          ADMIN103
      WRITE(ISOUT,1) L,N                                                ADMIN104
      WRITE(ISOUT,2)                                                    ADMIN105
      WRITE(ISOUT,3)                                                    ADMIN106
      WRITE(ISOUT,4)                                                    ADMIN107
 915  DO 920 K=1,KBHX                                                   ADMIN108
      UKNL=USUM(K,N,L)                                                  ADMIN109
      VKNL=VSUM(K,N,L)                                                  ADMIN110
      IF(ABS(UKNL)-1.0E-30) 9151,9154,9154                              ADMIN111
 9151 IF(ABS(VKNL)-1.0E-30) 9152,9153,9153                              ADMIN112
 9152 RKNL=0.                                                           ADMIN113
      GO TO 9155                                                        ADMIN114
 9153 RKNL=1.57079633                                                   ADMIN115
      GO TO 9155                                                        ADMIN116
```

```
9154 RKNL=ATAN(VKNL/UKNL)                                              ADMIN117
9155 DXKNL=DXSUM(K,N,L)                                                ADMIN118
     DYKNL=DYSUM(K,N,L)                                                ADMIN119
     IF(K-KBHX) 916,9156,920                                           ADMIN120
9156 RSUM(K,N,L)=RKNL                                                  ADMIN121
     GO TO 9195                                                        ADMIN122
916  ZSTEP=ZBH(K+1)-ZBH(K)                                             ADMIN123
     USUM(K,N,L)=UKNL*ZSTEP                                            ADMIN124
     VSUM(K,N,L)=VKNL*ZSTEP                                            ADMIN125
     RSUM(K,N,L)=RKNL*ZSTEP                                            ADMIN126
     DXSUM(K,N,L)=DXKNL*ZSTEP                                          ADMIN127
     DYSUM(K,N,L)=DYKNL*ZSTEP                                          ADMIN128
     DSUM=ZSTEP*DFZ(K,N,L)                                             ADMIN129
     M=K-1                                                             ADMIN130
     IF(M) 920,919,918                                                 ADMIN131
918  USUM(K,N,L)=USUM(K,N,L)+USUM(M,N,L)                               ADMIN132
     VSUM(K,N,L)=VSUM(K,N,L)+VSUM(M,N,L)                               ADMIN133
     RSUM(K,N,L)=RSUM(K,N,L)+RSUM(M,N,L)                               ADMIN134
     DXSUM(K,N,L)=DXSUM(K,N,L)+DXSUM(M,N,L)                            ADMIN135
     DYSUM(K,N,L)=DYSUM(K,N,L)+DYSUM(M,N,L)                            ADMIN136
919  DKAV=DKAV+DSUM                                                    ADMIN137
9195 IF(MC(1).LT.1) GO TO 920                                          ADMIN138
     WRITE(ISOUT,5) K,UKNL,VKNL,WFZ(K,N,L),DXKNL,DYKNL,DFZ(K,N,L),RKNL ADMIN139
920  CONTINUE                                                          ADMIN140
     IF(MC(1).NE.2) GO TO 9205                                         ADMIN141
     WRITE(ISOUT,6)                                                    ADMIN142
     WRITE(ISOUT,7) (K,USUM(K,N,L),VSUM(K,N,L),DXSUM(K,N,L),           ADMIN143
    2DYSUM(K,N,L),RSUM(K,N,L),K=1,KBHX)                                ADMIN144
9205 DKAV=DKAV/ZSPAN                                                   ADMIN145
     WRITE(ISOUT,8) L,N,      DKAV                                     ADMIN146
     CALL CNTR(NET,NETSU,N,XG,YG,ICF,JCF,NCF)                          ADMIN147
     XQ=XG                                                             ADMIN148
     YQ=YG                                                             ADMIN149
     CALL        NEST(NET,NETSU,XQ,YQ,NDATQ,XL,XR,YL,YU,ICF,JCF,NCF)   ADMIN150
     AREAN=(XR-XL)*(YU-YL)                                             ADMIN151
     DAVG(L)=DAVG(L)+DKAV*AREAN                                        ADMIN152
     DO 9210 KL=1,KBHX                                                 ADMIN153
9210 WAVG(KL,L)= WAVG(KL,L) + WFZ(KL,N,L)*AREAN                        ADMIN154
921  CONTINUE                                                          ADMIN155
     DAVG(L)=DAVG(L)/AREA                                              ADMIN156
     DO 9215 KL=1,KBHX                                                 ADMIN157
9215 WAVG(KL,L)=WAVG(KL,L) / AREA                                      ADMIN158
     WRITE(ISOUT,15) L,          DAVG(L)                               ADMIN159
     WRITE(ISOUT,16)                                                   ADMIN160
     WRITE(ISOUT,17)(K,WAVG(K,L),K=1,KBHX)                             ADMIN161
922  CONTINUE                                                          ADMIN162
     RETURN                                                            ADMIN163
     END                                                               ADMIN164
```

```
      SUBROUTINE ADVEC(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,
     1TDEP,CAV,PMAS,PSIZ,ICF,JCF,NCF,KBHF,NDATF,LTIMF,CAVS,WFZ)
C     SEPTEMBER, 1971
C     SUBROUTINE ADVEC TRANSPORTS PARCELS BY SIMPLE ADVECTION PLUS
C     SETTLING. PARCEL TOP AND BASE ARE TRANSPORTED SEPARATELY, AND THE
C     RESULTS ARE SMEARED. THE COMMON VARIABLE ZP IS REDEFINED HEREIN.
C     ZP    - PARCEL CENTER Z COORDINATE BEFORE ADVECTION, EXCEPT AS
C             REDEFINED IN SUB. ADVEC
      COMMON /QPARM/ IPOUT ,IPARIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM
     1,NSEQO ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT
     2,EDDY  ,FMBEL ,LSTEP ,MC(18),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP
     4,DINCR ,DOWN  ,TP    ,ZLOW  ,DWAF  ,RWAF  ,ROPART,ZUPP  ,VETA
     5,DOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP
      DIMENSION NET(ICF,JCF),NETSU(NCF),ZBH(KBHF),USUM(KBHF,NDATF,LTIMF)
      DIMENSION VSUM(KBHF,NDATF,LTIMF),DXSUM(KBHF,NDATF,LTIMF)
      DIMENSION DYSUM(KBHF,NDATF,LTIMF),TIMUP(LTIMF)
      DIMENSION RSUM(KBHF,NDATF,LTIMF)
      DIMENSION CAVS(KBHF),WFZ(KBHF,NDATF,LTIMF)
      MC3=MC(3)
      EPS=0.1
      NDEP=0
      ZDEP=ZMIN
CHANGE ZP FROM PARCEL CENTER TO PARCEL BASE ALTITUDE.
      ZP=ZLOW
CALCULATE TRANSPORT OF PARCEL BASE.
      IF (   (ZP-ZDEP).LE.EPS) GO TO 1411
      CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,
     1NDEP,TDEP,ZDEP,XOL,YOL,ZOL,TOL,SIGXL,SIGYL,ROL,NDATL,ICF,JCF,NCF,
     2KBHF,NDATF,LTIMF,0,CAVS,WFZ)
      GO TO 1412
 1411 TOL=TP
      XOL=XP
      YOL=YP
      ZOL=ZP
      ROL=0.
      SIGXL=RWAF
      SIGYL=RWAF
CHANGE ZP FROM PARCEL BASE TO PARCEL TOP ALTITUDE.
 1412 ZP=ZLOW+DWAF
CALCULATE TRANSPORT OF PARCEL TOP.
      IF( ZP-ZDEP .LE.EPS) GO TO 1414
      CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,
     1NDEP,TDEP,ZDEP,XOU,YOU,ZOU,TOU,SIGXU,SIGYU,ROU,NDATU,ICF,JCF,NCF,
     2KBHF,NDATF,LTIMF,0,CAVS,WFZ)
      GO TO 1415
 1414 TOU=TP
      XOU=XP
      YOU=YP
      ZOU=ZP
      ROU=0.
      SIGXU=RWAF
      SIGYU=RWAF
CALCULATE SHEAR OF PARCEL TOP AND BASE RESULTS.
 1415 ZOUTN=(ZOL+ZOU)/2.
      TOUTN=(TOL+TOU)/2.
      IF(ABS(XOU-XOL).GE.1.0E-30) GO TO 1404
      IF(ABS(YOU-YOL).GE.1.0E-30) GO TO 1403
```

```
      ROUTN=0.                                                          ADVEC 59
      GO TO 1405                                                        ADVEC 60
 1403 ROUTN=1.57079633                                                  ADVEC 61
      GO TO 1405                                                        ADVEC 62
 1404 ROUTN=ATAN((YOU-YOL)/(XOU-XOL))                                   ADVEC 63
 1405 R=ROUTN-ROL                                                       ADVEC 64
      SXL=1./SQRT((COS(R)/SIGXL)**2+(SIN(R)/SIGYL)**2)                  ADVEC 65
      SYL=1./SQRT((SIN(R)/SIGXL)**2+(COS(R)/SIGYL)**2)                  ADVEC 66
      R=ROUTN-ROU                                                       ADVEC 67
      SXU=1./SQRT((COS(R)/SIGXU)**2+(SIN(R)/SIGYU)**2)                  ADVEC 68
      SYU=1./SQRT((SIN(R)/SIGXU)**2+(COS(R)/SIGYU)**2)                  ADVEC 69
      SXOTN=(SXU+SXL+SQRT((XOU-XOL)**2+(YOU-YOL)**2))/2.                ADVEC 70
      SYOTN=SQRT(SYU*SYL)                                               ADVEC 71
      XOUTN=XOL+(SXOTN-SXL)*COS(ROUTN)                                  ADVEC 72
      YOUTN=YOL+(SXOTN-SXL)*SIN(ROUTN)                                  ADVEC 73
      CALL DUMPER(XOUTN,YOUTN,ZOUTN,TOUTN,SXOTN,SYOTN,PHAS,PSIZ,ROUTN,0,ADVEC 74
     1ISOUT,IPOUT,MC3,NBLK)                                             ADVEC 75
      RETURN                                                            ADVEC 76
      END                                                               ADVEC 77
```

```
      SUBROUTINE AMBNT(ZCH,DFZ,WFZ,ZHT,W,DIFF,AA,BB,CC,DENOM,E,F,NET,
     1NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,NTRIP,LTIM,ICF,JCF,NCF,
     2KBHF,NDATF,LTIMF,KKF,CAVS)
C     SEPTEMBER 1971
C     SUBROUTINE AMBNT DETERMINES FROM INPUT FALL RATE AND VERTICAL WIND
C     AND TURBULENCE DATA THE COEFFICIENT DATA VECTORS
C     AA(K)=S2*(DIFF(K+1)+DIFF(K))+S1*F(K+1),
C     BB(K)=S2*(DIFF(K+1)+2.*DIFF(K)+DIFF(K-1))+S1*F(K),
C     CC(K)=S2*(DIFF(K)+DIFF(K-1)),
C     DENOM(K)=1+THETA*(BB(K)-CC(K)*E(K-1)),
C     E(K)=THETA*AA(K)/DENOM(K),
C     WHERE
C     S1=DT/DZ,
C     S2=DT/(2.*(DZ)**2)),
C     AND F IS A TEMPORARY WORKING SPACE FOR VERTICAL VELOCITIES.
C     NOTE THAT Q(K) FOR THE NEXT TIME STEP IS GIVEN BY
C     AA(K)*Q(K+1)-BB(K)*Q(K)+CC(K)*Q(K-1).
C     DFKXS1- VERTICAL DIFFUSIVITY AT ALTITUDE INCREMENT KX-1
C     LTIM  - ATMOS. UPDATE INDEX FOR ARRAYS DFZ AND WFZ
C     NTRIP - OPTION CODE FOR HORIZONTAL ADVECTION OF MASS ALOFT
C             POSITIVE IF NDATO IS STORED IN NTRIP
C             NEGATIVE IF NDATO IS TO BE FOUND VIA CALL TO SUB. TRANP
      COMMON /OPARM/ IPOUT ,IPARIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM
     1,NSEQO ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT
     2,EDDY  ,FMBEL ,LSTEP ,MC(18),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP
     4,DINCR ,DOWN  ,TP    ,ZLOW  ,DWAF  ,PKAF  ,ROPART,ZUPP  ,VETA
     5,DOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP
      COMMON /DBLE/DFKXS1,EFFLUX,FMAB,PHI,THETO
      DOUBLE PRECISION AA(KKF),BB(KKF),CC(KKF),DENOM(KKF),E(KKF),F(KKF)
      DIMENSION DIFF(KKF),W(KKF),ZHT(KKF),DFZ(KBHF,NDATF,LTIMF)
      DIMENSION WFZ(KBHF,NDATF,LTIMF),TIMUP(LTIMF),ZBH(KBHF),ZCH(KBHF)
      DIMENSION  DXSUM(KBHF,NDATF,LTIMF),DYSUM(KBHF,NDATF,LTIMF)
      DIMENSION   USUM(KBHF,NDATF,LTIMF), VSUM(KBHF,NDATF,LTIMF)
      DIMENSION   RSUM(KBHF,NDATF,LTIMF),NET(ICF,JCF),NETSU(NCF)
      DIMENSION CAVS(KBHF)
      DOUBLE PRECISION DFKXS1,EFFLUX,FMAB,PHI,THETO
      DOUBLE PRECISION S1,S2
      DATA PROGRM/*AMBNT */
CONSTRUCT F AND DIFF FOR K=0,...,KX
      NDEP=100
      TDEP=TIME
      NDATO=IABS(NTRIP)
COMPUTE KBH AND INITIALIZE
      DIFF(KKM)=0.
      F(KKM)=0.
      ZOLD=ZHT(KKM)
      DOLD=DIFF(KKM)
      FOLD=F(KKM)
      DO 1 K=1,KBHX
      KBH=K
      ZNEW=ZCH(KBH)
      IF(ZOLD.LT.ZNEW) GO TO 2
    1 IF(KBH.EQ.KBHX) CALL ERROR(PROGRM,-1,ISOUT)
    2 IF(NTRIP.LT.-1)
     1CALL       TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,
     2NDEP,TDEP,ZNEW,XO,YO,ZO,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF,
     3NDATF,LTIMF,1,CAVS,WFZ)
```

```
      DNEW=DFZ(KBH,NDATO,LTIM)                                          AMBNT 59
      FNEW=-WFZ(KBH,NDATO,LTIM)                                         AMBNT 60
      DSLOPE=(DNEW-DOLD)/(ZNEW-ZOLD)                                    AMBNT 61
      FSLOPE=(FNEW-FOLD)/(ZNEW-ZOLD)                                    AMBNT 62
      DO 5 KK=KKM1,KKX                                                  AMBNT 63
      ZHTKK=ZHT(KK)                                                     AMBNT 64
      IF(ZHTKK.LT.ZNEW) GO TO 4                                         AMBNT 65
      IF(KBH.LT.KBHX-1) GO TO 3                                         AMBNT 66
   40 DIFF(KK)=DNEW                                                     AMBNT 67
      F(KK)=FNEW                                                        AMBNT 68
      GO TO 5                                                           AMBNT 69
    3 KBH=KBH+1                                                         AMBNT 70
      IF(ZHTKK.GE.ZCH(KBH)) GO TO 30                                    AMBNT 71
      DOLD=DNEW                                                         AMBNT 72
      FOLD=FNEW                                                         AMBNT 73
      ZOLD=ZNEW                                                         AMBNT 74
      GO TO 38                                                          AMBNT 75
   30 KBC=KBH+1                                                         AMBNT 76
      IF(KBC.LT.KBHX-1) GO TO 32                                        AMBNT 77
      KBH=KBC                                                           AMBNT 78
   31 ZNEW=ZCH(KBH)                                                     AMBNT 79
      IF(NTRIP.LT.-1)                                                   AMBNT 80
     1CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,  AMBNT 81
     2NDEP,TDEP,ZNEW,XO,YC,ZO,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF,  AMBNT 82
     3NDATF,LTIMF,1,CAVS,WFZ)                                           AMBNT 83
      DNEW=DFZ(KBH,NDATO,LTIM)                                          AMBNT 84
      FNEW=-WFZ(KBH,NDATO,LTIM)                                         AMBNT 85
      GO TO 40                                                          AMBNT 86
   32 KBHXM1=KBHX-1                                                     AMBNT 87
      DO 35 K=KBC,KBHXM1                                                AMBNT 88
      KBH=K                                                             AMBNT 89
      IF(ZHTKK.GE.ZCH(KBH)) GO TO 35                                    AMBNT 90
      ZOLD=ZCH(KBH-1)                                                   AMBNT 91
      IF(NTRIP.LT.-1)                                                   AMBNT 92
     1CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,  AMBNT 93
     2NDEP,TDEP,ZOLD,XO,YC,ZO,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF,  AMBNT 94
     3NDATF,LTIMF,1,CAVS,WFZ)                                           AMBNT 95
      DOLD=DFZ(KBH-1,NDATO,LTIM)                                        AMBNT 96
      FOLD=-WFZ(KBH-1,NDATO,LTIM)                                       AMBNT 97
      GO TO 38                                                          AMBNT 98
   35 CONTINUE                                                          AMBNT 99
      GO TO 31                                                          AMBNT100
   38 ZNEW=ZCH(KBH)                                                     AMBNT101
      IF(NTRIP.LT.-1)                                                   AMBNT102
     1CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,  AMBNT103
     2NDEP,TDEP,ZNEW,XO,YC,ZO,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF,  AMBNT104
     3NDATF,LTIMF,1,CAVS,WFZ)                                           AMBNT105
      DNEW=DFZ(KBH,NDATO,LTIM)                                          AMBNT106
      FNEW=-WFZ(KBH,NDATO,LTIM)                                         AMBNT107
      DSLOPE=(DNEW-DOLD)/(ZNEW-ZOLD)                                    AMBNT108
      FSLOPE=(FNEW-FOLD)/(ZNEW-ZOLD)                                    AMBNT109
    4 DIFF(KK)=DOLD+DSLOPE*(ZHTKK-ZOLD)                                 AMBNT110
      F(KK)=FOLD+FSLOPE*(ZHTKK-ZOLD)                                    AMBNT111
    5 CONTINUE                                                          AMBNT112
CORRECT DIFFUSIVITIES FOR CROSSING TRAJECTORIES EFFECTS                 AMBNT113
      DO 150 KK=KKM,KKX                                                 AMBNT114
  150 DIFF(KK)=DOWN*DIFF(KK)                                            AMBNT115
COMBINE STILL-AIR FALL RATES WITH VERTICAL WINDS                        AMBNT116
```

```
      DO 250 KK=KKM,KKX                                                 AMBNT117
  250 F(KK)=F(KK)+H(KK)                                                 AMBNT118
COMPUTE E(0) IN ACCORDANCE WITH REFLECTIVITY CONTROL VARIABLE VETA      AMBNT119
      E(KKM)=0.                                                         AMBNT120
      IF(VETA.LT.0.) GO TO 17                                           AMBNT121
      WREF=F(KKMA1)                                                     AMBNT122
      DREF=(DIFF(KKM)+DIFF(KKMA1))/(2.*DZ)                              AMBNT123
      E(KKM)=(WREF+DREF)/(VETA+DREF)                                    AMBNT124
COMPUTE AA,BB,CC,DENOM, AND E FOR K=1,...,KX-1                          AMBNT125
   17 S2=DZ                                                             AMBNT126
      S1=DT                                                             AMBNT127
      S1=S1/S2                                                          AMBNT128
      S2=S1/(2.*S2)                                                     AMBNT129
      DO 18 KK=KKMA1,KKX                                                AMBNT130
   18 CC(KK)=S2*(DIFF(KK)+DIFF(KK-1))                                   AMBNT131
      DO 19 KK=KKM,KKXS1                                                AMBNT132
   19 AA(KK)=CC(KK+1)+S1*F(KK+1)                                        AMBNT133
      DO 20 KK=KKMA1,KKXS1                                              AMBNT134
      BB(KK)=AA(KK-1)+CC(KK+1)                                          AMBNT135
      DENOM(KK)=1.+THETQ*(BB(KK)-CC(KK)*E(KK-1))                        AMBNT136
   20 E(KK)=THETQ*AA(KK)/DENOM(KK)                                      AMBNT137
      DFKXS1=DIFF(KKXS1)                                                AMBNT138
C     WRITE(ISOUT,2221)                                                 AMBNT139
C2221 FORMAT(*0CONTENTS OF ARRAYS ZHT, F, DIFF, AA, BB, CC, DENOM, E*)  AMBNT140
C     WRITE(ISOUT,2222) ZHT                                             AMBNT141
C     WRITE(ISOUT,2222) F                                               AMBNT142
C     WRITE(ISOUT,2222) DIFF                                            AMBNT143
C     WRITE(ISOUT,2222) AA                                              AMBNT144
C     WRITE(ISOUT,2222) BB                                              AMBNT145
C     WRITE(ISOUT,2222) CC                                              AMBNT146
C     WRITE(ISOUT,2222) DENOM                                           AMBNT147
C     WRITE(ISOUT,2222) E                                               AMBNT148
C2222 FORMAT(*0*(13E10.2))                                              AMBNT149
      RETURN                                                            AMBNT150
      END                                                               AMBNT151
```

```
      SUBROUTINE SPRVS(NET,NETSU,ZBH,ZCH,TIMUP,USUM,VSUM,DXSUM,DYSUM,    SPRVS  1
     1RSUM,DFZ,WFZ,DAVG,WAVG,ALT,RHO,ATEMP,AA,BB,CC,DENOM,DIFF,E,F,Q,W,  SPRVS  2
     2ZHT,ICF,JCF,NCF,KBHF,NDATF,LTIMF,KKF,NATF,CAVS)                    SPRVS  3
C     SEPTEMBER 1971                                                    SPRVS  4
C     SUBROUTINE SPRVS SUPERVISES DIFFUSIVE AND/OR ADVECTIVE TRANSPORT  SPRVS  5
C     OF FALLOUT PARCELS LISTED ON TAPE IPARIN. PARCEL PARAMETERS ARE   SPRVS  6
C     STORED IN ARRAYS XPAR,YPAR,ZPAR,TPAR,PDAM,PSAM,RWFR,DWFR,ZLWF,VWFRSPRVS  7
C     ONLY ONE PARCEL IS TRANSPORTED AT A TIME. FOR THIS PARCEL ABOVE   SPRVS  8
C     ITEMS ARE STORED IN XP,YP,ZP,TP,PSIZ,PMAS,RWAF,DWAF,ZLOW,VWAF.    SPRVS  9
C     XPAR  - X COORDINATE OF PARCEL CENTER DATA VECTOR (AT TIME TPAR)  SPRVS 10
C     YPAR  - Y COORDINATE OF PARCEL CENTER DATA VECTOR (AT TIME TPAR)  SPRVS 11
C     ZPAR  - Z COORDINATE OF PARCEL CENTER DATA VECTOR (AT TIME TPAR)  SPRVS 12
C     TPAR  - TIME OF DEFINITION OF CLOUD PARCEL DATA VECTOR            SPRVS 13
C     PDAM  - MIDPOINT OF PARCEL PARTICLE SIZE CLASS DATA VECTOR        SPRVS 14
C     PSAM  - TOTAL MASS OF PARCEL DATA VECTOR (AT TIME TPAR)           SPRVS 15
C     RWFR  - RADIUS OF PARCEL AT C. O. M. DATA VECTOR (AT TIME TPAR)   SPRVS 16
C     DWFR  - PARCEL THICKNESS DATA VECTOR (AT TIME TPAR)               SPRVS 17
C     ZLWF  - ALTITUDE OF PARCEL BASA DATA VECTOR (AT TIME TPAR)        SPRVS 18
C     VWFR  - PARCEL VOLUME DATA VECTOR (AT TIME TPAR)                  SPRVS 19
      COMMON /QPARM/ IPOUT ,IPARIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM      SPRVS 20
     1,NSEQO ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT    SPRVS 21
     2,EDDY  ,FMBEL ,LSTEP ,MC(18),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN     SPRVS 22
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP      SPRVS 23
     4,DINCR ,DOWN  ,TP    ,ZLOW  ,DWAF  ,RWAF  ,ROPART,ZUPP  ,VETA    SPRVS 24
     5,DOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP           SPRVS 25
      COMMON /QDBLE/DFKXS1,EFFLUX,FMAB,PHI,THETQ                        SPRVS 26
      DOUBLE PRECISION AA(KKF),BB(KKF),CC(KKF),DENOM(KKF),E(KKF),F(KKF) SPRVS 27
      DOUBLE PRECISION Q(KKF)                                           SPRVS 28
      DIMENSION ALT(NATF),RHO(NATF),ATEMP(NATF)                         SPRVS 29
      DIMENSION NET(ICF,JCF),NETSU(NCF),ZBH(KBHF),ZCH(KBHF),DIFF(KKF)   SPRVS 30
      DIMENSION USUM(KBHF,NDATF,LTIMF),VSUM(KBHF,NDATF,LTIMF),W(KKF)    SPRVS 31
      DIMENSION DXSUM(KBHF,NDATF,LTIMF),DYSUM(KBHF,NDATF,LTIMF)         SPRVS 32
      DIMENSION DFZ(KBHF,NDATF,LTIMF),ZHT(KKF)                          SPRVS 33
      DIMENSION WFZ(KBHF,NDATF,LTIMF),DAVG(LTIMF),WAVG(KBHF,LTIMF)      SPRVS 34
      DIMENSION TIMUP(LTIMF),RSUM(KBHF,NDATF,LTIMF)                     SPRVS 35
      DIMENSION XPAR(100),YPAR(100),ZPAR(100),TPAR(100),PDAM(100)       SPRVS 36
      DIMENSION PSAM(100),RWFR(100),DWFR(100),ZLWF(100),VWFR(100)       SPRVS 37
      DIMENSION CAVS(KBHF)                                              SPRVS 38
      DOUBLE PRECISION DFKXS1,EFFLUX,FMAB,PHI,THETQ                     SPRVS 39
 8014 FORMAT(*+*T102,E12.4,I4)                                          SPRVS 40
 8015 FORMAT(*+*T103,*AIRBORNE (ADVCN)*)                                SPRVS 41
 8016 FORMAT(* *I4,3E12.4)                                              SPRVS 42
 8017 FORMAT(*+*T103,*AIRBORNE (DIFFN)*)                                SPRVS 43
 8018 FORMAT(*+*T103,*ADVECTIVE TRNSPT*)                                SPRVS 44
 8019 FORMAT(*+*T103,* IMPACTED WAFER*)                                 SPRVS 45
 8020 FORMAT(*+*T103,*OUTSIDE WINDGRID*)                                SPRVS 46
 8021 FORMAT(*0*36X,*PARTICLE SIZE CLASS*E12.4,*  MICRONS*)             SPRVS 47
 8022 FORMAT(* *22X,*FALL RATE*E12.4,*  METERS/SEC  AT ALTITUDE*E12.4,* SPRVS 48
     1 METERS*/22X,*UPPER LIMIT INPUT ALTITUDE FOR ADV. TRANSP. IS*E12.4SPRVS 49
     2,* METERS*)                                                       SPRVS 50
 8024 FORMAT(*0*T2,*NSEQ*T11,*XP*T23,*YP*T35,*ZP*T47,*TP*T58,*PMAS*T70,*SPRVS 51
     1RWAF*T82,*ZLOW*T94,*DWAF*T107,*DZ*T115,*KX*/)                     SPRVS 52
 8025 FORMAT(*0NEGATIVE DEPOSIT. WAFER NO.*I4,* AT TIME*E12.4,*. VARIABLSPRVS 53
     5ES EFFLUX,FMAB,DEP *2D12.4,E12.4/)                                SPRVS 54
      DATA  PROGRM/*SPRVS */                                            SPRVS 55
      JF=100                                                            SPRVS 56
      KKMA1=KKM+1                                                       SPRVS 57
      THETQ=THETA                                                       SPRVS 58
```

```
      PHI=1.-THETQ                                                      SPRVS 59
      KXBF=0                                                            SPRVS 60
      MC3=MC(3)                                                         SPRVS 61
      NSEQ=0                                                            SPRVS 62
      PSZBE=-2.0                                                        SPRVS 63
COMPUTE OVERALL AVERAGE VERTICAL VELOCITY FOR THE FIRST UPDATE          SPRVS 64
      WAVGK=0.0                                                         SPRVS 65
      KBHM1=KBHX-1                                                      SPRVS 66
      DO 50 K=1,KBHM1                                                   SPRVS 67
   50 WAVGK=WAVGK + WAVG(K,1)*(ZBH(K+1) - ZBH(K))                       SPRVS 68
      WAVGK=WAVGK/(ZBH(KBHX)-ZBH(1))                                    SPRVS 69
COMPUTE TIMEX MARGIN FACTOR FOR ADVECTIVE TRANSPORT AIRBORNE TEST       SPRVS 70
      IF(NDATX-1)70,70,60                                               SPRVS 71
   60 SLOP=1.1                                                          SPRVS 72
      GO TO 80                                                          SPRVS 73
   70 SLOP=1.0                                                          SPRVS 74
COMPUTE MINIMUM SMALL ALTITUDE INCREMENT DZMIN                          SPRVS 75
   80 DZMIN=(ZMAX-ZMIN)/KX                                              SPRVS 76
CUE IPARIN TAPE AT BEGINNING OF INPUT PARCEL BLOCK                      SPRVS 77
 100  READ(IPARIN) NP                                                   SPRVS 78
      IF(NP.LE.0) GO TO 806                                             SPRVS 79
      IF(NP.GT.JF) CALL ERROR(PROGRM,-100,ISOUT)                        SPRVS 80
COPY IN A BLOCK OF INPUT PARCEL PARAMETERS FROM TAPE IPARIN             SPRVS 81
      READ(IPARIN) (XPAR(J),YPAR(J),ZPAR(J),TPAR(J),PDAM(J),PSAM(J),    SPRVS 82
     1RWFR(J),DWFR(J),ZLWF(J),VWFR(J),J=1,NP)                           SPRVS 83
COMMENCE PROCESSING BLOCK OF INPUT PARCELS ONE AT A TIME                SPRVS 84
      DO 1000 J=1,NP                                                    SPRVS 85
      NSEQ=NSEQ+1                                                       SPRVS 86
      IF(NSEQ.LT.NSEQO) GO TO 1000                                      SPRVS 87
      XP=XPAR(J)                                                        SPRVS 88
      YP=YPAR(J)                                                        SPRVS 89
      ZP=ZPAR(J)                                                        SPRVS 90
      TP=TPAR(J)                                                        SPRVS 91
      PSIZ=1.0E6*PDAM(J)                                                SPRVS 92
      PMAS=PSAM(J)                                                      SPRVS 93
      RWAF=RWFR(J)/2.                                                   SPRVS 94
      DWAF=DWFR(J)                                                      SPRVS 95
      ZLOW=ZLWF(J)                                                      SPRVS 96
      VWAF=VWFR(J)                                                      SPRVS 97
CHECK FOR NEW PARTICLE SIZE CLASS                                       SPRVS 98
      IF(ABS((PSIZ-PSZBE)/PSIZ).LE.1.0E-10) GO TO 103                   SPRVS 99
      WRITE(ISOUT,8021) PSIZ                                            SPRVS100
COMPUTE MID-ATMOSPHERE FALL RATE FAV FOR NEW PARTICLE SIZE CLASS        SPRVS101
      H=(ZMIN+ZMAX)/2.                                                  SPRVS102
      CALL TRPL(H,NAT,ALT,RHO,DEN)                                      SPRVS103
      CALL TRPL(H,NAT,ALT,ATEMP,VIS)                                    SPRVS104
      CALL  FALRT(PSIZ,ROPART,H,DEN,VIS,FAV,ISOUT)                      SPRVS105
      FAV=FAV-WAVGK                                                     SPRVS106
COMPUTE UPPER LIMIT ALTITUDE FOR ADVECTIVE TRANSPORT OF THIS SIZE PART. SPRVS107
      CALL CALIB(ZBH,KBHX,ZMIN,-1,KBHZ)                                 SPRVS108
      CALL TRPL(ZBH(KBHZ),NAT,ALT,ATEMP,VIS)                            SPRVS109
      CALL TRPL(ZBH(KBHZ),NAT,ALT,RHO,DEN)                              SPRVS110
      CALLFALRT(PSIZ,ROPART,ZBH(KBHZ),DEN,VIS,CAV,ISOUT)                SPRVS111
      TDEP=TP + (ZBH(KBHZ) - ZMIN) /(CAV - WAVG(KBHZ-1,1))              SPRVS112
      KBHM1=KBHX-1                                                      SPRVS113
      DO 1001 IZ=KBHZ,KBHM1                                             SPRVS114
      CALL TRPL(ZCH( IZ ),NAT,ALT,ATEMP,VIS)                            SPRVS115
      CALL TRPL(ZCH( IZ ),NAT,ALT,RHO,DEN)                              SPRVS116
```

```
      CALLFALRT(PSIZ,ROPART,ZCH( IZ ),DEN,VIS,CAV,ISOUT)                SPRVS117
      TDEP=TDEP + (ZBH(IZ+1) - ZBH(IZ))/(CAV- WAVG(IZ,1))               SPRVS118
      IF(TDEP.GT.SLOP*TIMEX) GO TO 1002                                 SPRVS119
 1001 CONTINUE                                                          SPRVS120
      ZLIM=5.0E4                                                        SPRVS121
      GO TO 1003                                                        SPRVS122
 1002 ZLIM=ZBH(IZ+1)                                                    SPRVS123
 1003 WRITE(ISOUT,8022) FAV,H,ZLIM                                      SPRVS124
      WRITE(ISOUT,8024)                                                 SPRVS125
COMPUTE PARTICLE FALL RATE TABLE FOR EACH ATMOSPHERIC STRATUM WHEN      SPRVS126
C     A NEW PARTICLE SIZE IS ENCOUNTERED                                SPRVS127
      DO 101 KKZ=1,KBHX                                                 SPRVS128
      CALL TRPL(ZCH(KKZ),NAT,ALT,ATEMP,VIS)                             SPRVS129
      CALL TRPL(ZCH(KKZ),NAT,ALT,RHO,DEN)                               SPRVS130
 101  CALL FALRT(PSIZ,ROPART,ZCH(KKZ),DEN,VIS,CAVS(KKZ),ISOUT)          SPRVS131
COMPUTE DIFFUSIVITY CORRECTIONS FOR NEW PARTICLE SIZE CLASS             SPRVS132
      DOWN=(FAV*EDDY)**2                                                SPRVS133
      CROSS=1./SQRT(1.+4.*DOWN)                                         SPRVS134
      DOWN=1./SQRT(1.+DOWN)                                             SPRVS135
      PSZBE=PSIZ                                                        SPRVS136
 103  WRITE(ISOUT,8016) NSEQ,XP,YP,ZP,TP,PMAS,RWAF,ZLOW,DWAF            SPRVS137
CANCEL PROCESSING OF PARCEL IF IT HAS ALREADY IMPACTED                  SPRVS138
      IF (IFIX(DWAF).GT.0) GO TO 1200                                   SPRVS139
      WRITE(ISOUT,8019)                                                 SPRVS140
      CALL        DUMPER(XP,YP,ZP,TP, RWAF, RWAF, PMAS,PSIZ,0.,0,       SPRVS141
     1ISOUT,IPOUT,MC3,NBLK)                                             SPRVS142
      GO TO 1000                                                        SPRVS143
COMPUTE INDEX OF MESH OR SUB-MESH CONTAINING PARCEL CENTER POSITION     SPRVS144
 1200 CALL NEST(NET,NETSU,XP,YP,NDATP,XL,XR,YL,YU,ICF,JCF,NCF)          SPRVS145
CANCEL PROCESSING OF PARCEL IF IT IS INPUT OUTSIDE ATMOS.               SPRVS146
      IF(NDATP.GT.0) GO TO 1248                                         SPRVS147
      WRITE(ISOUT,8020)                                                 SPRVS148
      GO TO 1000                                                        SPRVS149
COMPUTE AVERAGE FALL RATE CAV                                           SPRVS150
 1248 ZUPP=ZLOW+DWAF                                                    SPRVS151
      ZLO=ZLOW-ZMIN                                                     SPRVS152
      ZUP=ZUPP-ZMIN                                                     SPRVS153
      H=ZMIN+ZUP/2.                                                     SPRVS154
      CALL TRPL(H,NAT,ALT,RHO,DEN)                                      SPRVS155
      CALL TRPL(H,NAT,ALT,ATEMP,VIS)                                    SPRVS156
      CALL  FALRT(PSIZ,ROPART,H,DEN,VIS,CAV,ISOUT)                      SPRVS157
CANCEL PROCESSING OF PARCEL IF IT WILL REMAIN AIRBORNE BY DIFFUSION     SPRVS158
      DAV=DOWN*DAVG(1)                                                  SPRVS159
      CAV=CAV-WAVGK                                                     SPRVS160
      TFLY=TIMEX-TP                                                     SPRVS161
      IF(TFLY.LE.0.) GO TO 1249                                         SPRVS162
      CALL ESTM(ZUP,CAV,DAV,TFLY,PUP)                                   SPRVS163
      CALL ESTM(ZLO,CAV,DAV,TFLY,PLO)                                   SPRVS164
      PDEST=1.0-(PUP-PLO)/(ZUP-ZLO)                                     SPRVS165
      IF(CSKIP.LE.PDEST) GO TO 1250                                     SPRVS166
 1249 WRITE(ISOUT,8017)                                                 SPRVS167
      GO TO 1000                                                        SPRVS168
COMPUTE TRANSPORT BY ADVECTION IF TRUNCATION ERROR IS EXCEEDED          SPRVS169
 1250 IF(DZMIN.LE.0.2*DAV/FAV) GO TO 1500                               SPRVS170
CANCEL PROCESSING OF PARCEL IF IT WILL REMAIN AIRBORNE BY ADVECTION     SPRVS171
      IF(ZLOW.LT.ZLIM) GO TO 1409                                       SPRVS172
      WRITE(ISOUT,8015)                                                 SPRVS173
      GO TO 1000                                                        SPRVS174
```

```
 1409 WRITE(ISOUT,8018)                                                 SPRVS175
      CALL        ADVEC(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,  SPRVS176
     1TDEP,CAV,PMAS,PSIZ,ICF,JCF,NCF,KBHF,NDATF,LTIMF,CAVS,WFZ)          SPRVS177
      GO TO 1000                                                        SPRVS178
COMPUTE SMALL ALTITUDE INCREMENT DZ                                     SPRVS179
 1500 DZMAX=DWAF                                                        SPRVS180
 1501 IF(DZMAX.GE.DZMIN) GO TO 1502                                     SPRVS181
      DZMAX=2.0*DZMAX                                                   SPRVS182
      GO TO 1501                                                        SPRVS183
 1502 DZ=0.2*DAV/FAV                                                    SPRVS184
      IF(DZ.GT.DZMAX) DZ=DZMAX                                          SPRVS185
      KX=(ZMAX-ZMIN)/DZ+1.                                              SPRVS186
      IF(KX.LT.KXMIN) KX=KXMIN                                          SPRVS187
      DZ=(ZMAX-ZMIN)/KX                                                 SPRVS188
      WRITE(ISOUT,8014) DZ,KX                                           SPRVS189
COMPUTE ALTITUDE INCREMENT AND FALL RATE DATA VECTORS                   SPRVS190
      IF(KX.EQ.KXBE) GO TO 1305                                         SPRVS191
      KKX=KKM+KX                                                        SPRVS192
      KKXS1=KKX-1                                                       SPRVS193
      DO 1304 KK=1,KKX                                                  SPRVS194
      ZHT(KK)=ZMIN+DZ*(KK-KKM)                                          SPRVS195
      CALL TRPL(ZHT(KK),NAT,ALT,RHO,DEN)                                SPRVS196
      CALL TRPL(ZHT(KK),NAT,ALT,ATEMP,VIS)                              SPRVS197
 1304 CALL  FALRT(PSIZ,ROFAPT,ZHT(KK),DEN,VIS,W(KK),ISOUT)              SPRVS198
      KXBE=KX                                                           SPRVS199
COMPOSE INITIAL CONCENTRATION DATA VECTOR                               SPRVS200
 1305 CALL        CONC(ZHT,Q,KKF)                                       SPRVS201
CANCEL PROCESSING OF WAFER IF INITIAL AIRBORNE MASS IS INADEQUATE       SPRVS202
      IF(FMAB.LT.1) GO TO 1000                                          SPRVS203
      DEPBE=0.                                                          SPRVS204
      NTRIP=NDATP                                                       SPRVS205
COMPUTE FIRST DEPOSIT INCREMENT TIME INTERVAL TDELT                     SPRVS206
      TDELT=TFLY/NDELT                                                  SPRVS207
      TLARG=TDELT                                                       SPRVS208
      TAE=4.*ZUP/(3.*CAV)                                               SPRVS209
      IF(TAE.LT.TFLY) TDELT=TAE/NDELT                                   SPRVS210
      TSMAL=MINT*DT                                                     SPRVS211
      IF(TDELT.LT.TSMAL) TDELT=TSMAL                                    SPRVS212
COMMENCE DEPOSIT TIME LOOP                                              SPRVS213
      TIME=TP                                                           SPRVS214
      TPAUS=TIME                                                        SPRVS215
      LTIM=-1                                                           SPRVS216
      NPASS=1                                                           SPRVS217
      XOBE=XP                                                           SPRVS218
      YOBE=YP                                                           SPRVS219
    1 LSTEP=TDELT/DT +1.                                                SPRVS220
      TDELT=LSTEP*DT                                                    SPRVS221
  135 TPAUS=TPAUS+TDELT                                                 SPRVS222
COMPUTE DATA SET TIME INDEX LTIM                                        SPRVS223
      CALL CALI9(TIMUP,LTIMX,TIME,+1,LTIMA)                             SPRVS224
      IF(LTIM.NE.LTIMA) GO TO 3                                         SPRVS225
      IF(IABS(NTRIP)-1) 31,32,31                                        SPRVS226
    3 LTIM=LTIMA                                                        SPRVS227
COMPOSE CONCENTRATION COEFFICIENT DATA ARRAYS                           SPRVS228
   31 CALL        AMBNT(ZCH,DFZ,WFZ,ZHT,W,DIFF,AA,BB,CC,DENOM,E,F,NET,   SPRVS229
     1NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,NTRIP,LTIM,ICF,JCF,NCF,SPRVS230
     2KBHF,NDATF,LTIMF,KKF,CAVS)                                        SPRVS231
      NTRIP=-NDATX                                                      SPRVS232
```

```
COMPUTE NUMBER OF ITERATIONS LSTEP OF SOLUTION TO VERTICAL DIFFUSION     SPRVS233
C     DIFFERENCE EQUATION FOR THIS DEPOSIT INCREMENT TIME INTERVAL       SPRVS234
   32 IF(LTIM.GE.LTIMX) GO TO 138                                        SPRVS235
      IF(TPAUS.LT.TIMUP(LTIM+1)) GO TO 140                               SPRVS236
      TPAUS=TIMUP(LTIM+1)                                                SPRVS237
      TDELT=TPAUS-TIME                                                   SPRVS238
      LSTEP=TDELT/DT+1.                                                  SPRVS239
      TDELT=LSTEP*DT                                                     SPRVS240
  138 IF(TPAUS.LT.TIMEX) GO TO 140                                       SPRVS241
      TPAUS=TIMEX                                                        SPRVS242
      TDELT=TPAUS-TIME                                                   SPRVS243
      LSTEP=TDELT/DT+1.                                                  SPRVS244
      TDELT=LSTEP*DT                                                     SPRVS245
      IF(LSTEP.LE.0) GO TO 1000                                          SPRVS246
  140 IF(NPASS.NE.1) GO TO 4                                             SPRVS247
      NPASS=2                                                            SPRVS248
      TDELTP=TDELT                                                       SPRVS249
COMPUTE DEPOSIT INCREMENT FRACTIONAL MASS DEP                            SPRVS250
    4 CALL        DIFFF(Q,AA,BB,CC,DENOM,E,F,DEP,KKF)                    SPRVS251
CHECK DEPOSIT INCREMENT FRACTIONAL MASS DEP AGAINST DOPEN                SPRVS252
      IF(DEP.GE.-DOPEN*TDELT) GO TO 5                                    SPRVS253
      WRITE(ISOUT,8025) NSEQ,TIME,EFFLUX,FMAB,DEP                        SPRVS254
    5 IF(DEP.GE.DOPEN*TDELT) GO TO 7                                     SPRVS255
CHECK CUMULATIVE AIRBORNE FRACTIONAL MASS FMAB AGAINST FMBEL             SPRVS256
      IF(SNGL(FMAB).GT.FMBEL) GO TO 135                                  SPRVS257
      GO TO 1000                                                         SPRVS258
COMPARE RATE OF CHANGE OF DEPOSITION RATE DPDP WITH DINCR AND ADJUST     SPRVS259
C     NEXT TDELT                                                         SPRVS260
    7 DPDP=((DEP/TDELT)-(DEPBE/TDELTB))/TDELT                            SPRVS261
      IF(DPDP.LT.DINCR) GO TO 10                                         SPRVS262
      TDELTB=TDELT                                                       SPRVS263
      TDELT=TDELT/2.                                                     SPRVS264
      IF(TDELT.LT.TSMAL) TDELT=TSMAL                                     SPRVS265
      GO TO 13                                                           SPRVS266
   10 IF(DPDP.GT.DINCR) GO TO 13                                         SPRVS267
      TDELTB=TDELT                                                       SPRVS268
      TDELT=2.*TDELT                                                     SPRVS269
      IF(TDELT.GT.TLARG) TDELT=TLARG                                     SPRVS270
COMPUTE DEPOSIT INCREMENT MASS PMDEP                                     SPRVS271
   13 PMDEP=PMAS*DEP                                                     SPRVS272
      DEPBE=DEP                                                          SPRVS273
      NDEP=0                                                             SPRVS274
      ZDEP=ZMIN                                                          SPRVS275
      TDEP=TIME                                                          SPRVS276
COMPUTE DEPOSIT INCREMENT POSITION (XO,YO,ZO) AND HORIZONTAL DISPERSION  SPRVS277
C     PARAMETERS (SIGXO,SIGYO,RO) AT TIME=TO                             SPRVS278
      CALL        TRANP(NET,NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,PSUM,  SPRVS279
     1NDEP,TDEP,ZDEP,XO,YO,ZO,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF,  SPRVS280
     2NDATF,LTIMF,1,CAVS,WF7)                                            SPRVS281
CONTINUE ON TO NEXT WAFER IF THIS ONE LIES OUTSIDE WINDFIELD             SPRVS282
      IF(NDATO.LE.0) GO TO 1000                                          SPRVS283
COLLECT FINAL RESULTS FOR THIS WAFER AND STORE IN BUFFER DATA VECTORS    SPRVS284
      XH=(XO+XOBE)/2.                                                    SPRVS285
      YH=(YO+YOBE)/2.                                                    SPRVS286
      CALL        DUMPER(XH,YH,ZO,TO,SIGXO,SIGYO,PMDEP,PSIZ,RO,0,        SPRVS287
     1ISOUT,IPOUT,MC3,NBLK)                                              SPRVS288
      XOBE=XO                                                            SPRVS289
      YOBE=YO                                                            SPRVS290
```

```
      IF(TIME+DT.LT.TIMEX) GO TO 1                                      SPRVS291
 1000 CONTINUE                                                          SPRVS292
      GO TO 100                                                         SPRVS293
COPY OUT BUFFER DATA VECTORS. WAFER PROCESSING HAS BEEN COMPLETED       SPRVS294
  806 CALL        DUMPER(0.,0.,0.,0.,   0.,   0.,   0.,  0.,0.,999,     SPRVS295
     1ISOUT,IPOUT,MC3,NBLK)                                             SPRVS296
      CALL        DUMPER(0.,0.,0.,0.,   0.,   0.,   0.,  0.,0.,999,     SPRVS297
     1ISOUT,IPOUT,MC3,NBLK)                                             SPRVS298
      REWIND IPARIN                                                     SPRVS299
      END FILE IPOUT                                                    SPRVS300
      REWIND IPOUT                                                      SPRVS301
      RETURN                                                            SPRVS302
      END                                                               SPRVS303
```

```
      SUBROUTINE TRANP(NET.NETSU,ZBH,TIMUP,USUM,VSUM,DXSUM,DYSUM,RSUM,   TRANP  1
     1NDEP,TDEP,ZDEP,XO,YC,ZC,TO,SIGXO,SIGYO,RO,NDATO,ICF,JCF,NCF,KBHF, TRANP  2
     2NDATF,LTIMF,KRIP,CAVS,WFZ)                                        TRANP  3
C     SEPTEMBER 1971                                                    TRANP  4
C     SUBROUTINE TRANP DETERMINES (AT AN INPUT TERMINAL ALTITUDE PLANE) TRANP  5
C     THE WAFER HORIZONTAL CENTER POSITION AND DISPERSION PARAMETERS FORTRANP  6
C     AN INPUT TRANSPORT FLIGHT TIME                                    TRANP  7
C     CAVS   - PARTICLE FALL RATE TABULATED FOR EACH ALTITUDE STRATUM   TRANP  8
C     DXSUM - TURBULENCE X COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY   TRANP  9
C     DYSUM - TURBULENCE Y COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY   TRANP 10
C     NET   - HORIZONTAL SPACE CONTROL NET MESH 2-DIM. ARRAY            TRANP 11
C     NETSU - HORIZONTAL SPACE CONTROL NET SUB-MESH DATA VECTOR         TRANP 12
C     RSUM  - WIND HEADING ORIENTATION ANGLE (WEIGHTED SUM) 3-DIM. ARRAYTRANP 13
C     USUM  - WIND X COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY         TRANP 14
C     VSUM  - WIND Y COMPONENT (WEIGHTED SUM) 3-DIM. DATA ARRAY         TRANP 15
C     ZBH   - ATMOSPHERE STRATA BASE-ALTITUDE DATA VECTOR               TRANP 16
C     MODE  - COMPUTATION MODE SWITCH                                   TRANP 17
C                0 RAPID COMPUTATION                                    TRANP 18
C                1 LAYERWISE COMPUTATION                                TRANP 19
C     NDEP  - OPTION CONTROL VARIABLE                                   TRANP 20
C                ZERO IF SIGXO AND SIGYO ARE TO BE COMPUTED             TRANP 21
C                NON-ZERO IF SIGXO AND SIGYO ARE NOT TO BE COMPUTED AND TRANP 22
C                IF NDATO IS TO BE POSITIVE ALWAYS                      TRANP 23
C     TDEP  - ADVECTIVE TRANSPORT TIME INTERVAL                         TRANP 24
C     KRIP   - CONTROL VARIABLE                                         TRANP 25
C                0 FOR ADVECTIVE TRANSPORT                              TRANP 26
C                1 FOR DIFFUSIVE TRANSPORT                              TRANP 27
C     WFZ   - VERTICAL WIND FIELD                                       TRANP 28
C     ZDEP  - ADVECTIVE TRANSPORT TERMINAL ALTITUDE                     TRANP 29
C     TO    - TIME AFTER PARCEL ADVECTION                               TRANP 30
C     XO    - PARCEL CENTER X COORDINATE AFTER ADVECTION                TRANP 31
C     YO    - PARCEL CENTER Y COORDINATE AFTER ADVECTION                TRANP 32
C     ZO    - PARCEL CENTER Z COORDINATE AFTER ADVECTION                TRANP 33
C     SIGXO - PARCEL MASS HOR. STAND. DEV. DOWNWIND AFTER ADVECTION     TRANP 34
C     SIGYO - PARCEL MASS HOR. STAND. DEV. CROSSWIND AFTER ADVECTION    TRANP 35
C     NDATO - HORIZONTAL INDEX OF LATTICE CELL CONTAINING POINT (XC,YC) TRANP 36
C     RO    - WIND HEADING ORIENTATION ANGLE AFTER ADVECTION            TRANP 37
      COMMON /OPARM/ IPOUT ,IPADIN,NBLK  ,NAT   ,NDELT ,KX    ,KKM      TRANP 38
     1,MSEQD ,ICX   ,JCX   ,NCX   ,KBHX  ,NDATX ,LTIMX ,ISIN  ,ISOUT    TRANP 39
     2,EDDY  ,FMDEL ,LSTEP ,MC(18),WINT  ,XLLC  ,YLLC  ,THETA ,ZMIN     TRANP 40
     3,CSKIP ,MINT  ,ZMAX  ,TIMEX ,DT    ,DZ    ,XP    ,YP    ,ZP      TRANP 41
     4,DINCR ,DCWN  ,TP    ,ZLOW  ,DWAF  ,RWAF  ,RDPART,ZUPP  ,VETA    TRANP 42
     5,MOPEN ,CROSS ,TIME  ,KKMA1 ,KKX   ,KKXS1 ,KXMIN ,NDATP          TRANP 43
      DIMENSION NET(ICF,JCF),NETSU(NCF),ZBH(KBHF),USUM(KBHF,NDATF,LTIMF)TRANP 44
      DIMENSION VSUM(KBHF,NDATF,LTIMF),DXSUM(KBHF,NDATF,LTIMF)          TRANP 45
      DIMENSION DYSUM(KBHF,NDATF,LTIMF),TIMUP(LTIMF)                    TRANP 46
      DIMENSION CAVS(KBHF)                                              TRANP 47
      DIMENSION RSUM(KBHF,NDATF,LTIMF)                                  TRANP 48
      DIMENSION WFZ(KBHF,NDATF,LTIMF)                                   TRANP 49
      DATA PROGRM/*TRANP */                                             TRANP 50
    2 FORMAT(* TIME=*E12.4,*. ALT=*E12.4,*. X-POS=*E12.4,*. Y-POS=*E12.4TRANP 51
     2,*. CELL=*I6,* REACHED*)                                          TRANP 52
    3 FORMAT(* TIME=*E12.4,*. ALT=*E12.4,*. X-POS=*E12.4,*. Y-POS=*E12.4TRANP 53
     3,*. CELL=*I6,* ATTEMPTED*)                                        TRANP 54
    4 FORMAT(*0WAFER WITH INITIAL CONFIGURATION XP,YP,ZP,TP *4E12.4/*REQTRANP 55
     1QUIRED CHANNELLING AT CONFIGURATION XC,YC,ZC,TC *4E12.4)          TRANP 56
      EPSILO=.0005                                                      TRANP 57
      EPS=EPSILO*WINT                                                   TRANP 58
```

```
      EPST=EPSILO*TDEP                                                  TRANP 59
      EPS7=.1                                                           TRANP 60
      XO=XP                                                             TRANP 61
      YO=YP                                                             TRANP 62
      ZO=ZP                                                             TRANP 63
      TO=TP                                                             TRANP 64
      SIGXO=0.                                                          TRANP 65
      SIGYO=0.                                                          TRANP 66
      RO=0.                                                             TRANP 67
      NDATO=NDATP                                                       TRANP 68
      NDTC1=0                                                           TRANP 69
      NDTO1=0                                                           TRANP 70
      KBHC1=0                                                           TRANP 71
      KBHO1=0                                                           TRANP 72
 1000 CONTINUE                                                          TRANP 73
      KAY=-1                                                            TRANP 74
      IF (KRIP.EQ.1) GO TO 50                                           TRANP 75
CALCULATE FALL VELOCITY FOR ADVECTION ON BASIS OF LOCAL                 TRANP 76
C     FALL RATE AND WIND FIELD  AND STORE IN WBAR                       TRANP 77
      CALL CALIB(ZBH,KBHX,ZO,KAY,KBHZ)                                  TRANP 78
      CALL CALIB(TIMUP,LTIMX,TO,1,LTIM)                                 TRANP 79
      WBAR=WFZ(KBHZ-1,NDATO,LTIM)-CAYS(KBHZ-1)                          TRANP 80
      IF (WBAR) 112,111,110                                             TRANP 81
COMPUTE THE VERTICAL PSEUDO-VELOCITY WBAR AND STORE ITS SIGN IN KAY     TRANP 82
C     FOR THE DIFFUSIVE SETTLING CASE                                   TRANP 83
   50 WBAR=(ZDEP-ZP)/(TDEP-TP)                                          TRANP 84
      IF(WBAR) 112,111,110                                              TRANP 85
  110 KAY=KAY+1                                                         TRANP 86
  111 KAY=KAY+1                                                         TRANP 87
  112 CONTINUE                                                          TRANP 88
CALIBRATION OF ZDEP AGAINST ZBH YIELDS TERMINAL ZBH-PLANE               TRANP 89
      IF (KRIP.EQ.1) CALL CALIB(ZBH,KBHX,ZDEP,KAY,KBH)                  TRANP 90
CONSIDER KAY=0 CASE INDEPENDENTLY                                       TRANP 91
  200 IF(KAY.NE.0) GO TO 206                                            TRANP 92
      IF (KRIP.EQ.1) GO TO 205                                          TRANP 93
C     IN THE ADVECTIVE TRANSPORT CASE WHENEVER THE ACTUAL FALL RATE     TRANP 94
C     IS ZERO THEN SET THE DEPOSITION TIME INCREMENT EQUAL TO THE       TRANP 95
C     TIME LEFT BEFORE  THE WIND FIELD IS UPDATED                       TRANP 96
      TSEG=TIMUP(LTIM+1)-TO                                             TRANP 97
      KBHC=KBHZ+1                                                       TRANP 98
      KBHO=KBHZ                                                         TRANP 99
      GO TO 300                                                         TRANP100
  205 TSEG=TDEP-TO                                                      TRANP101
      MODE=1                                                            TRANP102
      KBHC=KBH                                                          TRANP103
      KBHO=KBHC-1                                                       TRANP104
      GO TO 300                                                         TRANP105
CALIBRATION OF ZO AGAINST ZBH YIELDS CURRENT ZBH-PLANE                  TRANP106
  206 CALL CALIB(ZBH,KBHX,ZO,-KAY,KTRY)                                 TRANP107
      CALL CALIB(ZBH,KBHX,ZO,+KAY,KBHC)                                 TRANP108
      IF (KRIP.EQ.0).OR.(KBHO.NE.KTRY)) GO TO 213                       TRANP109
CONSIDER EXCURSION TO TERMINAL ZBH-PLANE                                TRANP110
      KBHC=KBH                                                          TRANP111
      ZEST=ZBH(KBHC)                                                    TRANP112
      MODE=0                                                            TRANP113
      IF(KAY*(KBHO-KBHC)+1) 221,213,218                                 TRANP114
CONSIDER EXCURSION BETWEEN ADJACENT ZBH-PLANES                          TRANP115
  213 KBHC=KBHC+KAY                                                     TRANP116
```

```
      ZEST=ZBH(KBHC)                                                    TRANP117
      MODE=1                                                            TRANP118
      IF((KBHC.LE.KBHX).AND.(KAY*ZEST.LE.KAY*ZDEP)) GO TO 221           TRANP119
CONSIDER EXCURSION TO TERMINAL ZDEP-PLANE                               TRANP120
 218  KBHC=KBHO+KAY                                                     TRANP121
      ZEST=ZDEP                                                         TRANP122
      MODE=1                                                            TRANP123
  221 TSEG=(ZEST-ZO)/WBAR                                               TRANP124
CHECK IF UPDATE TIME BOUNDARY WILL BE CROSSED                           TRANP125
 300  TC=TO+TSEG                                                        TRANP126
      CALL CALIB(TIMUP,LTIMX,TO,1,LTIM)                                 TRANP127
      IF((LTIM.LT.LTIMX).AND.(TIMUP(LTIM+1).LE.TC))TSEG=TIMUP(LTIM+1)-TOTRANP128
COMPUTE AVERAGE HORIZONTAL VELOCITIES UBAR AND VBAR                     TRANP129
      KBHA=KBHO                                                         TRANP130
      KBHB=KBHC                                                         TRANP131
      IF(KAY.LT.0) GO TO 405                                            TRANP132
      KBHA=KBHC                                                         TRANP133
      KBHB=KBHO                                                         TRANP134
  405 CALL GETDA( USUM,ZBH,KBHA,KBHB,NDATO,LTIM, UBAR,KBHF,NDATF,LTIMF) TRANP135
      CALL GETDA( VSUM,ZBH,KBHA,KBHB,NDATO,LTIM, VBAR,KBHF,NDATF,LTIMF) TRANP136
 407  IF(NDEP.NE.0) GO TO 412                                           TRANP137
COMPUTE AVERAGE HORIZONTAL DISPERSION AND WIND ORIENTATION ANGLE        TRANP138
      CALL GETDA(DXSUM,ZBH,KBHA,KBHB,NDATO,LTIM,DXBAR,KBHF,NDATF,LTIMF) TRANP139
      CALL GETDA(DYSUM,ZBH,KBHA,KBHB,NDATO,LTIM,DYBAR,KBHF,NDATF,LTIMF) TRANP140
      CALL GETDA( RSUM,ZBH,KBHA,KBHB,NDATO,LTIM, RBAR,KBHF,NDATF,LTIMF) TRANP141
      RC=RO+RBAR                                                        TRANP142
      SIGXC=SIGXO+DXBAR*TSEG                                            TRANP143
      SIGYC=SIGYO+DYBAR*TSEG                                            TRANP144
COMPUTE CURRENT POSITION AND TIME (XC,YC,ZC,TC)                         TRANP145
 412  TC=TO+TSEG                                                        TRANP146
      ZC=ZO+WBAR*TSEG                                                   TRANP147
      XC=XO+UBAR*TSEG                                                   TRANP148
      YC=YO+VBAR*TSEG                                                   TRANP149
      CALL NEST(NET,NETSU,XC,YC,NDATC,XL,XR,YL,YU,ICF,JCF,NCF)          TRANP150
      IF(MC(4).EQ.1) WRITE(ISOUT,3) TC,ZC,XC,YC,NDATC                   TRANP151
COMPARE CURRENT MESH INDEX NDATC WITH PREVIOUS MESH INDEX NDATO         TRANP152
      IF(NDATC.EQ.NDATO) GO TO 700                                      TRANP153
COMPUTE INTERPOLATED POINT                                              TRANP154
      XT=XC                                                             TRANP155
      YT=YC                                                             TRANP156
      ZT=ZC                                                             TRANP157
      IF (NODE.EQ.0) GO TO 213                                          TRANP158
      CALL BOUN(NET,NETSU,XT,YT,XO,YO,XC,YC,ICF,JCF,NCF)                TRANP159
      ZC=SQRT((XT-XC)**2+(YT-YC)**2)/((XT-XO)**2+(YT-YO)**2)            TRANP160
      ZC=ZT+ZC*(ZO-ZT)                                                  TRANP161
      IF(ABS(WBAR).LE.1.0E-30) GO TO 510                                TRANP162
      TSEG=(ZC-ZO)/WBAR                                                 TRANP163
      GO TO 518                                                         TRANP164
 510  IF(ABS(UBAR).LE.1.0E-30) GO TO 513                                TRANP165
      TSEG=(XC-XO)/UBAR                                                 TRANP166
      GO TO 518                                                         TRANP167
 513  IF(ABS(VBAR).LE.1.0E-30) GO TO 516                                TRANP168
      TSEG=(YC-YO)/VBAR                                                 TRANP169
      GO TO 518                                                         TRANP170
 516  CALL ERROR(PROGRM,516,ISOUT)                                      TRANP171
      RETURN                                                            TRANP172
 518  IF(NDEP.NE.0) GO TO 521                                           TRANP173
      RC=RO+RBAR                                                        TRANP174
```

```
      SIGXC=SIGXO+DXBAR*TSEG                                            TRANP175
      SIGYC=SIGYO+DYBAR*TSEG                                            TRANP176
  521 TC=TO+TSEG                                                        TRANP177
      CALL NEST(NET,NETSU,XC,YC,NDATC,XL,XR,YL,YU,ICF,JCF,NCF)          TRANP178
CHECK IF PARCEL CENTER POSITION IS OSCILLATING                          TRANP179
      IF((KBHO1.NE.KBHO).OR.(KBHC1.NE.KBHC).OR.(NDTC1.NE.NDATC).OR.     TRANP180
     1(NDTO1.NE.NDATO)) GO TO 626                                       TRANP181
      IF(HC(4).EQ.1) WRITE(ISOUT,4) XP,YP,ZP,TP,XC,YC,ZC,TC             TRANP182
      CALL CNTR(NET,NETSU,NDATO,XG,YG,ICF,JCF,NCF)                      TRANP183
      XO=XG                                                             TRANP184
      YO=YG                                                             TRANP185
      CALL NEST(NET,NETSU,XO,YO,NDATO,XLO,XRO,YLO,YUO,ICF,JCF,NCF)      TRANP186
CLEAR STORED MESH AND STRATUM INDICES                                   TRANP187
      NDTC1=0                                                           TRANP188
      NDTO1=0                                                           TRANP189
      KBHC1=0                                                           TRANP190
      KBHO1=0                                                           TRANP191
CHANNEL WAFER CENTER POSITION ALONG APPROPRIATE CELL BOUNDARY           TRANP192
      SPE=2.*EPS                                                        TRANP193
      IF((ABS(XLO-XR).GT.SPE).AND.(ABS(XRO-XL).GT.SPE)) GO TO 616       TRANP194
      UBAR=0.                                                           TRANP195
      CALL GETDA( VSUM,ZBH,KBHA,KBHB,NDATO,LTIM,UBARC,KBHF,NDATF,LTIMF) TRANP196
      IF(ABS(UBARC).LE.ABS(UBAR)) GO TO 407                             TRANP197
      UBAR=UBARC                                                        TRANP198
      NDATO=NDATC                                                       TRANP199
      GO TO 407                                                         TRANP200
  616 IF((ABS(YLO-YU).GT.SPE).AND.(ABS(YUO-YL).GT.SPE))                 TRANP201
     1CALL ERROR(PROGPM,616,ISOUT)                                      TRANP202
      UBAR=0.                                                           TRANP203
      CALL GETDA( USUM,ZBH,KBHA,KBHB,NDATO,LTIM,UBARC,KBHF,NDATF,LTIMF) TRANP204
      IF(ABS(UBARC).LE.ABS(UBAR)) GO TO 407                             TRANP205
      UBAR=UBARC                                                        TRANP206
      NDATO=NDATC                                                       TRANP207
      GO TO 407                                                         TRANP208
COMMIT PREVIOUS AND CURRENT MESH AND STRATUM INDICES TO STORAGE         TRANP209
  626 NDTC1=NDATO                                                       TRANP210
      NDTO1=NDATC                                                       TRANP211
      KBHC1=KBHC                                                        TRANP212
      KBHO1=KBHO                                                        TRANP213
CONVERT XO,YO,ZO,TO,SIGXO,SIGYO, AND NDATO TO CURRENT VALUES            TRANP214
  700 ZO=ZC                                                             TRANP215
      XO=XC                                                             TRANP216
      YO=YC                                                             TRANP217
      TO=TC                                                             TRANP218
      NDATO=NDATC                                                       TRANP219
      IF(HC(4).EQ.1) WRITE(ISOUT,2) TO,ZC,XO,YO,NDATO                   TRANP220
      IF(NDEP.NE.0) GO TO 709                                           TRANP221
      SIGXO=SIGXC                                                       TRANP222
      SIGYO=SIGYC                                                       TRANP223
      RO=RC                                                             TRANP224
CHECK IF CURRENT POSITION IS OUTSIDE ATMOSPHERE                         TRANP225
  708 IF(NDATO.LE.0) GO TO 710                                          TRANP226
      IF (KRIP.EQ.1) GO TO 709                                          TRANP227
C     IF DEPOSITION PLANE IS REACHED OR TRANSPORT TIME LIMIT IS EXCEEDEDTRANP228
C     EXIT FROM TRANP, OTHERWISE RETURN TO TOP                          TRANP229
      IF ((   (ZO-ZDEP).LE.EPS2).OR.(   SQINSX-(G).LE.EPST)) GO TO 720  TRANP230
      GO TO 1000                                                        TRANP231
  709 CONTINUE                                                          TRANP232
```

```
CHECK IF TOTAL FLIGHT TIME HAS BEEN EXHAUSTED                          TRANP233
      IF(TO+EPST-TDEP) 200,720,720                                     TRANP234
CARRY PARCEL CENTER BACK INTO ATMOS. IF NDEP IS NOT EQUAL TO ZERO      TRANP235
 710  IF(NDEP.EQ.0) GO TO 720                                          TRANP236
      XO=XO-2.*EPS                                                     TRANP237
      YO=YO-2.*EPS                                                     TRANP238
      CALL NEST(NET,NETSU,XO,YO,NDATO,XL,YR,YL,YU,ICF,JCF,NCF)         TRANP239
      IF(NDATO.GT.0) GO TO 720                                         TRANP240
 714  XO=XO+4.*EPS                                                     TRANP241
      YO=YO+4.*EPS                                                     TRANP242
      CALL NEST(NET,NETSU,XO,YO,NDATO,XL,YR,YL,YU,ICF,JCF,NCF)         TRANP243
      IF(NDATO.LE.0) CALL ERROR(PROGRM,720,ISOUT)                      TRANP244
COMPUTE HORIZ. DISPERSION IF NDEP IS NOT EQUAL TO ZERO                 TRANP245
  720 IF(NDEP.NE.0) RETURN                                             TRANP246
      R2=RWAF**2                                                       TRANP247
      IF(MC(10).EQ.1) GO TO 721                                        TRANP248
      SIGXO=2.*DOWN*SIGXO                                              TRANP249
      SIGYO=2.*CROSS*SIGYO                                             TRANP250
      GO TO 722                                                        TRANP251
  721 TRIP=TO-TP                                                       TRANP252
      DSPRTX=SIGXO/TRIP                                                TRANP253
      TONEX=4.*((R2/DSPRTX)**(1./3.))                                  TRANP254
      IF(TRIP.LE.TONEX) SIGXO=DSPRTX*TONEX*(TRIP**2)/3.                TRANP255
      IF(TRIP.GT.TONEX) SIGXO=DSPRTX*(TRIP**3)/3.                      TRANP256
      SIGXO=SIGXO*(DOWN**(3./2.))                                      TRANP257
      DSPRTY=SIGYO/TRIP                                                TRANP258
      TONEY=4.*((R2/DSPRTY)**(1./3.))                                  TRANP259
      IF(TRIP.LE.TONEY) SIGYO=DSPRTY*TONEY*(TRIP**2)/3.                TRANP260
      IF(TRIP.GT.TONEY) SIGYO=DSPRTY*(TRIP**3)/3.                      TRANP261
      SIGYO=SIGYO*(CROSS**(3./2.))                                     TRANP262
  722 SIGXO=SQRT(R2+SIGXO)                                             TRANP263
      SIGYO=SQRT(R2+SIGYO)                                             TRANP264
      RETURN                                                           TRANP265
      END                                                              TRANP266
```