NASA CONTRACTOR REPORT

NASA CR-2631

NASA CR-2631

NASA/MSFC MULTILAYER DIFFUSION MODELS AND COMPUTER PROGRAMS - VERSION 5

R. K. Dumbauld and J. R. Bjorklund

Prepared by
H. E. CRAMER COMPANY, INC.
Salt Lake City, Utah 84108
for George C. Marshall Space Flight Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • DECEMBER 1975

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA CR-2631	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
4. TITLE AND SUBTITLE NASA/MSFC Multilayer Diffusion Models and Computer Programs — Version 5		5. REPORT DATE December 1975
		6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) R. K. Dumbauld and J. R. Bjorklund		8. PERFORMING ORGANIZATION REPORT # M155
9. PERFORMING ORGANIZATION NAME AND ADDRESS H. E. Cramer Company, Inc. P. O. Box 8049 Salt Lake City, Utah 84108		10. WORK UNIT NO. TR 75-301-08
		11. CONTRACT OR GRANT NO. NAS8-29033
		13. TYPE OF REPORT & PERIOD COVERED Contractor
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D. C. 20546		14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES
Prepared under the technical monitorship of Dr. J. B. Stephens, Aerospace Environment Division, Space Sciences Laboratory, NASA/MSFC

16. ABSTRACT

The purpose of this report is to document the latest developments in the transport and diffusion models and algorithms developed for use by NASA in predicting concentrations and dosages downwind from normal and abnormal launches of rocket vehicles and the associated computer programs for use in performing the calculations. This report consists of:

- A description of the mathematical specifications and procedures used in the Preprocessor Program to calculate rocket exhaust cloud rise, cloud dimensions and other input parameters to the transport and diffusion models
- A description of the revised mathematical specifications for the NASA/MSFC Multilayer Diffusion Models
- Descriptions of the Preprocessor and Version 5 of the NASA/MSFC Multilayer Diffusion Models Programs
- Users' instructions for implementing the Preprocessor and Multilayer Diffusion Models Programs
- Worked example problems illustrating the use of the models and computer programs

17. KEY WORDS Atmospheric Diffusion, Multilayer Diffusion Model, Rocket Exhaust Cloud Rise, Rocket Motor Exhaust Diffusion, Space Shuttle Environmental Studies	18. DISTRIBUTION STATEMENT Category 45

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	317	$9.25

For sale by the National Technical Information Service, Springfield, Virginia 22161

FOREWORD

This final report is submitted to Aerospace Environment Division, Space Sciences Laboratory, Science and Engineering Directorate, NASA-Marshall Space Flight Center, Alabama, in partial fulfillment of requirements under Contract No. NAS8-29033.

The H. E. Cramer Company, Inc., is indebted to Dr. Briscoe Stephens, Atmospheric Diffusion/Environmental Assessment Task Team, Aerospace Environment Division, for the technical guidance and useful suggestions in planning the revisions to the NASA/MSFC Multilayer Diffusion Models and in the development of the Preprocessor Program concept. The work under this contract is under the direction of Dr. Harrison E. Cramer, H. E. Cramer Company, Inc.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

SECTION 1

INTRODUCTION

1.1 BACKGROUND

Work under two concurrent NASA contracts (NAS8-21453 and NAS8-30503) resulted in publication of the NASA Handbook for Estimating Toxic Fuel Hazards (Dumbauld, Bjorklund, Cramer and Record, 1970). The handbook was accompanied by a computer program (NASA/MSFC Multilayer Diffusion Models Program) specifically designed for research-planning estimates of downwind dispersions from normal and abnormal launches of rocket vehicles as well as from accidental cold spills and leaks of toxic fuels. The transport and diffusion models that formed the basis of the program were extensions of the generalized concentration, dosage and deposition model concept originally developed for the U. S. Army (Cramer, et al., 1964; 1967) to include a multilayer tropospheric construct with provision for changes in atmospheric structure along the cloud trajectory downwind from the launch site. As experience was gained in the application of the NASA/MSFC Multilayer Diffusion Models Program, revisions were made to simplify its use and to incorporate the latest available advances in diffusion technology. Version 2 of the Program, designed for use in operational support of vehicle launches, was completed in 1973. A technical report by Dumbauld, Bjorklund and Bowers (1973) describes the updated transport, diffusion and cloud rise models and the operation of the computer program.

Since 1973, work has continued on the development and refinement of the models and the computer program.

1.2 PURPOSE

The purpose of this report is to document:

- The latest refinements and developments in the transport and diffusion models

- Version 5 of the NASA/MSFC Multilayer Diffusion Models Program
- The Preprocessor Program used in conjunction with Version 5 of the Multilayer Program

The majority of the changes in the NASA/MSFC Multilayer Diffusion Models Program have been made to further simplify the use of the models and computer program so that users can apply the techniques for hazard estimation with only a minimum knowledge of the technical aspects of diffusion meteorology and associated disciplines. The simplification of program use was accomplished in two ways. First, a Preprocessor Program was designed to automate the calculation of the:

- Effective height of the stabilized exhaust cloud
- Distribution of exhaust products in the meteorological layers containing the cloud
- Meteorological inputs to the transport and diffusion models

Use of the Preprocessor Program essentially requires only surface meteorological and radiosonde data be available for input. Second, the NASA/MSFC Multilayer Diffusion Models Program was redesigned for more efficient operation and an extensive capability for automated plotting of isopleths and centerline profiles of concentration, dosage and deposition due to gravitational settling and precipitation scavenging is now provided to assist users in interpretation of computational results.

In addition to simplifications in the use of computer programs, technical features of the transport and diffusion models have been altered to more accurately reflect observations of cloud rise during actual launches of solid-fueled vehicles and the physical aspects of the transport and diffusion process. For example, alterations have been made in assumptions regarding the physical shape of the ground cloud at time of cloud stabilization and in the diffusion models to account for partial reflection (or partial absorption) of material at air-surface interfaces.

1.3 ORGANIZATION OF THE REPORT

The main body of the report contains five sections. Section 2 contains a description of the models and algorithms contained in the Preprocessor Program. The generalized transport and diffusion models for calculating concentration, dosage and deposition contained in the NASA/MSFC Multilayer Diffusion Models Program are described in Section 3. A brief description of the Preprocessor and Diffusion Models Programs is given in Section 4 and Section 5 contains some sample problems and their solutions obtained using the Programs.

There are five appendices to the report. Appendix A contains user instructions for implementing the Preprocessor Program and Appendix B contains user instructions for implementing the NASA/MSFC Multilayer Diffusion Models Program. Computer program listings for both the programs are given in Appendix C. Appendix D contains example computer output listings for the sample problems described in Section 5 of the main text. Finally, a derivation of the vertical term for Model 4 of the NASA/MSFC Multilayer Diffusion Models Program is presented in Appendix E.

SECTION 2

PREPROCESSOR MODELS AND ALGORITHMS

A Preprocessor Program has been developed to simplify the calculation of source and meteorological inputs required by the NASA/MSFC Multilayer Diffusion Models Program. The models and algorithms incorporated in the Preprocessor Program for calculating the requisite model inputs are described in this section of the report. In its present form, the Preprocessor Program has the capability of processing data for the Space Shuttle, Titan III, Delta-Thor and Minuteman II vehicles.

2.1 FUEL PROPERTIES AND VEHICLE RISE DATA

Properties of the vehicle fuel and rise data are required to calculate cloud rise and the source strength distribution of atmospheric pollutants in the troposphere resulting from normal and abnormal launches. Two types of abnormal launches have been hypothesized for the Space Shuttle and Titan III vehicles. In one situation, it is assumed that a single solid engine of the Titan or Space Shuttle zero stage ignites and burns over the normal engine firing period while the vehicle remains in a hold-down configuration on the pad (single-engine burn). In the second pad-abort situation, it is assumed that an on-pad explosion ruptures the casings of the two solid engines and that all the solid propellant then falls to the ground in the vicinity of the pad and burns at a constant rate over a 5-minute period (slow burn). In both pad-abort situations, it is further assumed that the other vehicle stages are unaffected by the burning of the zero-stage solid propellant and do not therefore contribute in any way to the combustion products or heat released during the on-pad aborts. The net affects of this latter assumption is to minimize the cloud rise and maximize the concentration of the pollutants produced by the burning of the solid propellant. Slow-burn incidents in which fuel burns over a 5-minute period are also hypothesized for the Delta-Thor and Minuteman II vehicles. In the case of the Delta-Thor, both the solid propellant from the six castor engines and liquid fuel from the zero-stage engine are assumed to contribute to the heat available for cloud rise and to pollutant concentrations. The solid propellant from all three stages of the Minuteman II is assumed to fall to the ground near the launch site as the vehicle is destroyed just after clearing the launch silo.

Fuel expenditure and heat content data incorporated in the Preprocessor Program for normal and abnormal launches of the four types of vehicles are given in Table 2-1. Fuel expenditure rates for normal launches were obtained by averaging fuel consumption over the approximate period from lift-off until the vehicle is about 3 kilometers above the surface. For single-engine burns, the fuel expenditure rate was calculated by dividing the total propellant weight in a single engine by the normal firing period shown in the table. Similarly for slow burns in the vicinity of the launch area, the fuel expenditure rate was obtained by dividing the total weight of all solid engines, and the zero stage liquid engine in case of the Delta-Thor, by the total assumed burn time of 300 seconds.

The fuel heat contents used in calculating cloud rise are estimates based on the best available information and are subject to change as additional information becomes available. In case of the Tital III vehicle, the fuel heat content is based on estimates from two-phase (gas and solid) flow accounting approximately for heat gains from afterburning and heat losses due to radiation, and on the experience to date in predicting cloud rise for a number of actual launches using the instantaneous cloud rise model described in Section 2.2 below. Lacking better information, the same heat content has also been used for the solid engines of the Space Shuttle vehicle. A heat content value of 500 calories per gram was used for the liquid fuel in the zero stage engines of the Space Shuttle and Delta-Thor vehicles because previous experience in predicting cloud rise from Saturn launches indicates this value may be appropriate. The heat contents of the solid fuel for the Delta-Thor and Minuteman II are based on early estimates of the heat content for solid fueled engines and are probably low. However, experience in predicting cloud rise from a limited number of Delta-Thor

TABLE 2-1

FUEL EXPENDITURE AND HEAT CONTENT DATA

Property	Vehicle Type			
	Space Shuttle	Titan III	Delta-Thor	Minuteman II

(a) Normal Launch

Fuel Expenditure Rate (g sec^{-1})				
Solid Engine	9.385×10^6	4.174×10^6	6.05×10^5	3.771×10^5
Liquid Engine	1.531×10^6		3.13×10^5	
Effective Fuel Heat Content (cal g^{-1})				
Solid Fuel	2500	2500	691	691
Liquid Fuel	500		500	

(b) Single Engine Burn

Fuel Expenditure Rate (g sec^{-1})	3.753×10^6	1.742×10^6		
Normal Firing Period (sec)	122	122		
Effective Fuel Heat Content (cal g^{-1})	1274	1036		

TABLE 2-1

FUEL EXPENDITURE AND HEAT CONTENT DATA
(CONTINUED)

(c) Slow Burn

Property	Vehicle Type: Space Shuttle	Titan III	Delta-Thor	Minuteman II
Fuel Expenditure Rate (g sec^{-1})				
Solid Engine	3.052×10^6	1.301×10^6	7.462×10^4	2.87×10^7
Liquid Engine			2.212×10^5	
Total Burn Time (sec)	300	300	300	300
Effective Fuel Heat Content (cal g^{-1})	1000	1000	1000	1000

launches using the heat contents in Table 2-1 and a combination of the instantaneous and continuous cloud rise models indicate serious errors are not being made in the rise prediction. In any case, the conservative low estimates of heat available for cloud rise for the Delta-Thor and Minuteman II vehicles have the effect of minimizing cloud rise and maximizing ground-level concentrations. There is no experience in predicting cloud rise from launch aborts of any of the four vehicle types. The heat contents shown in Table 2-1 for single-engine burns of the Space Shuttle and Titan vehicles are based on the estimate that about 1500 calories per gram are available from the fuel burn from a single engine, but that some of the available heat is dissipated in heating and vaporizing 1.26×10^3 kilograms per second of water used to spray the launch pad during the burn. The heat content of 1000 calories per gram hypothesized for slow burns of fuel from the zero-stages of the vehicle is thought to be a realistic estimate of the heat available for cloud rise from the burning of unconfined fuel.

The fraction by weight of pollutants comprising the rocket exhaust products of the four types of vehicles used in the calculation of the vertical distribution of pollutants in the lower troposphere are shown in Table 2-2. Two sets of fractions are given for the Minuteman II because in an abnormal launch all three stages are hypothesized as being destroyed and in normal launches only the first stage contributes to the pollutant distribution in the lower troposphere. Recent indications (Cicerone, Stedman and Stolarski, 1973) are that CO in the exhaust plume may quickly oxidize to CO_2, in which case no toxic CO problem exists. However, because the oxidation of CO to CO_2 has not been verified by direct measurements, the distribution of CO in the lower troposphere is calculated in the Preprocessor Program.

It should be noted that the values of heat content and fuel expenditure rates in Tabel 2-1 and the fraction of pollutants by weight in Table 2-2 may differ from the values assigned as constants in the Preprocessor Program described in Appendix A. This occurs because, in the Program, the constants are manipulated in different ways to calculate cloud rise and the vertical distribution of pollutants. Thus, after the

TABLE 2-2

POLLUTANT COMPOSITION OF FUEL (Fraction by Weight)

Pollutant	Vehicle Type				
	Space Shuttle	Titan III	Delta-Thor	Minuteman II	
				Normal	Abnormal
HCl	0.207	0.210	0.208	0.197	0.204
CO*					
Solid Engine	0.280	0.279	0.223	0.220	0.219
Liquid Engine			0.473		
CO_2		0.029			
Al_2O_3	0.304	0.304	0.378	0.277	0.280

*May be oxidized to CO_2

TABLE 2-3

VEHICLE RISE DATA

Power-Law Coefficients	Vehicle Type			
	Space Shuttle	Titan III	Delta-Thor	Minuteman II
a	0.664	0.635	1.321	0.440
b	0.485	0.484	0.395	0.479

calculations are completed by the Preprocessor Program, the values shown in Tables 2-1 and 2-2 have been, in effect, used in the calculations.

The altitude-time curves of the various vehicles are also required to calculate the buoyant rise of the ground cloud of exhaust products. A simple power-law relationship is sufficiently accurate to describe the altitude-time curve in the first several thousand meters near the surface. A logarithmic least-squares regression curve of the form

$$\text{Time} = a\,(\text{Altitude})^{b} \tag{2-1}$$

where time is in seconds and altitude in meters was fitted to the altitude-time information for all vehicles. The resulting values of the coefficients a and b obtained from the fitting procedure are given in Table 2-3.

2.2 CLOUD RISE FORMULAS

The burning of rocket engines during launches and on-pad aborts results in the formation of a cloud of hot exhaust products which subsequently rises and entrains ambient air until an equilibrium with ambient conditions is reached. For normal launches, the cloud is formed principally by the forced ascent of hot, turbulent exhaust products that have been deflected laterally and vertically by the launch pad hardware and the ground surface. In the case of normal launches of solid-fueled vehicles or vehicles with a number of solid boosters, vehicle hold-down times are minimal and vehicle residence times in the lowest kilometer of the atmosphere are relatively short. The exhaust products contained in the stabilized ground cloud are therefore emitted over a time period of from about 10 to 30 seconds. Experience to date shows that the buoyant rise from vehicles with solid-fueled first stages is best predicted by using a cloud-rise model for instantaneous sources and rise from vehicles having liquid-fueled first stages is best predicted using a cloud-rise model for continuous sources. Thus, an instantaneous source cloud-rise model is used in the Preprocessor Program to predict buoyant rise of the ground cloud for normal launches of the Space Shuttle, Titan III and Minuteman II vehicles. Limited experience in predicting the buoyant rise of the ground cloud generated by normal launches of the Delta-Thor vehicle with its large liquid-fueled first stage and six solid-fueled boosters indicates that an average of the rise predicted using the instantaneous and continuous source cloud rise models is appropriate. While no cloud rise data are available for on-pad aborts of any of the four vehicle types, cloud rise data from static-tests of liquid fueled rocket engines indicate that the use of a cloud rise model for continuous sources is appropriate in these cases.

The instantaneous and continuous cloud-rise models used in the Preprocessor Program are based on work by Briggs (1969, 1970). Derivations of these models also appear in the report by Dumbauld, et al. (1973). Only the simplified forms of the models used in the Preprocessor Program are given below.

2.2.1 Instantaneous Source Cloud-Rise Model

The maximum cloud rise z_{mI} from an instantaneous source in a thermally stable atmosphere is given by the expression

$$z_{mI} = \left[\frac{8 \; F_I}{\gamma_I^3 \; s} \right]^{1/4} \qquad (2\text{-}2)$$

where

F_I = initial buoyancy term

$$= \frac{3 \, g \, Q_I}{4 \, c_p \, T \, \pi \, \rho}$$

g = gravitational acceleration (9.8 m sec^{-2})

Q_I = effective heat released (cal)

c_p = specific heat of air at constant pressure (cal g^{-1} ^oK)

T = ambient air temperature (^oK) at the surface reference height z_R

ρ = density of ambient air (g m^{-3}) at z_R

$$s = \frac{g}{T}\frac{\partial\Phi}{\partial z} \approx \frac{g}{T}\frac{\Delta\Phi}{\Delta z}$$

$\frac{\Delta\Phi}{\Delta z}$ = vertical gradient of ambient virtual potential temperature

γ_I = entrainment constant = 0.64

The time t_{Ik} required for the center of the ground cloud to reach an altitude z_k is given by the expression

$$t_{Ik} = s^{-1/2} \arccos\left(1 - \frac{(z_k)^4}{K}\right);\ \frac{(z_k)^4}{K} > 2 \qquad (2\text{-}3)$$

where

$$K = \frac{3\, Q_I}{\rho\, c_p\, \pi\, \gamma_I^3\, \frac{\Delta\,\Phi}{\Delta\, z}}$$

The time t^* required for the cloud to reach the stabilization height z_{mI} is thus given by the expression

$$t^* = \pi\, s^{-1/2} \qquad (2\text{-}4)$$

In calculating z_{mI} from Equation (2-2) the effective heat released for cloud rise Q_I is calculated from the relationship

$$Q_I = Q_F\, t_R\, \{z_{mI}\} \qquad (2\text{-}5)$$

where

Q_F = rate of heat released by the burning fuel

= H · W

H = fuel heat content

W = fuel expenditure rate

$t_R \{z_{mI}\}$ = time in seconds for the vehicle to reach the height of the centroid of the stabilized ground cloud z_{mI}

$$= a\,(z_{mI})^b \tag{2-6}$$

where a and b are the power-law coefficients from Table 2-3.

According to the above formulas, the following quantities are interrelated: the calculated maximum cloud rise z_{mI}, the height over which the virtual potential temperature gradient $\Delta\Phi / \Delta z$ is measured, and the value of $t_R\{z_{mI}\}$ used in obtaining Q_I. The final value of maximum cloud rise must therefore be found through iteration of Equation (2-1). The Preprocessor Program accepts radiosonde data which provides the temperature, pressure and relative humidity profiles as functions of height in the troposphere and the surface density ρ for use in the cloud rise calculation. The virtual potential temperature Φ_k at each radiosonde measurement height z_k is obtained from the expression (Tabata, 1973)

$$\Phi_k = T_k \left[\frac{1 + 1.61\, w_k}{1 + w_k}\right] \left[\frac{1000}{P_k}\right]^{0.288}$$

where

T_k = Air temperature at z_k in degrees absolute

P_k = barometric pressure at z_k (mb)

w_k = mixing ratio at z_k

$$= \frac{0.622\,(RH)_k\, e_{s;k}}{P_k - (RH)_k\, e_{s;k}}$$

$(RH)_k$ = relative humidity at z_k expressed as a fraction

$$e_{s;\,k} = \text{saturation vapor pressure at } z_k$$

$$= 10^{(c-dx-ex^2)}$$

$$x = 1000/T_k$$

and c, d and e are constants given by

$$c = 8.42926604$$

$$d = 1.82717843$$

$$e = 0.071208271$$

The iteration process then begins by assuming that the first level above the surface reference height ($z_k\ \{k = 1\} = z_R$) at which a radiosonde measurement is available ($z_k\ \{k = 2\}$) is equal to z_{mI} and solving Equation (2-6) for $t_R \left\{ z_{mI} = z_2 \right\}$ and Equation (2-5) for Q_I. The lapse rate of virtual potential temperature between the surface reference height and the first height above the surface is then obtained from the expression

$$\frac{\Delta\ \Phi}{\Delta\ z} = \frac{\Phi_k \{k = 2\} - \Phi_k \{k = 1\}}{z_k \{k = 2\} - z_R}$$

and Equation (2-2) solved for z_{mI}. If the value of z_{mI} calculated from Equation (2-6) is less than the value $z_k\ \{k = 2\}$, then 10 meters is subtracted from $z_k \{k = 2\}$ and the process repeated using the same value of $\Delta\Psi/\Delta z$ until the estimated value of z_{mI} is within $\pm$ 10 meters of the value of z_{mI} calculated from Equation (2-2). If on the first iteration step the value of z_{mI} calculated from Equation (2-6) exceeds the value $z_k\{k = 2\}$, then the value of $z_k\ \{k = 3\}$ is used as an estimate of z_{mI} and the iteration process is continued by increasing k until the value of z_{mI} from Equation (2-6) exceeds the estimated value of z_{mI}. However, for each iteration where $k > 2$, a least-square regression fit of the form

$$\frac{\Delta \Phi}{\Delta z} = \frac{\sum_{k=1}^{K} \left\{ \left[z_k - \left(\sum_{k=1}^{K} z_k / K \right) \right] \left[\Phi_k - \left(\sum_{k-1}^{K} \Phi_k / K \right) \right] \right\}}{\sum_{k=1}^{K} \left[z_k - \left(\sum_{k=1}^{K} z_k / K \right) \right]^2} \tag{2-7}$$

is used, at the suggestion of Dr. Briscoe Stephens of NASA/MSFC, to obtain an estimate of the vertical potential temperature gradient. As a result of this iteration procedure, the Preprocessor Program obtains an estimated final height of the cloud centroid within $\pm$ 10 meters of the exact value that should be calculated.

It should be noted that Equation (2-2) is for use when the atmosphere is thermally stable. For this reason, if the virtual potential temperature gradient is ever less than 3.322×10^{-4} degrees per meter, the gradient is set equal to 3.322×10^{-4} degrees per meter. Experience has shown that use of this device yields nearly the same final cloud rise as the cloud rise that would have been obtained had a formula for an adiabatic atmosphere been used in the calculation.

2.2.2 Continuous Source Cloud Rise Model

The maximum cloud rise z_{mc} from a continuous source is

$$z_{mc} = \left(\frac{6 \; F_c}{\bar{u}_c \, \gamma_c^{2} \, s} \right)^{1/3} \tag{2-8}$$

where

$$F_c = \text{buoyancy flux}$$

$$= \frac{g \; Q_c}{\pi \, \rho \, c_p \; T}$$

$$Q_c = \text{effective heat rate (cal sec}^{-1}\text{)}$$

$$= \quad H \bullet W$$

γ_c = entrainment constant = 0.5

$\bar{u}_c$ = height-weighted mean wind speed between the surface and z_{mc}

$$\approx \frac{\sum_{k=1}^{K} (\bar{u}_k) + \bar{u}_j}{z_{mc} - z_1}$$

$$\bar{u}_k = \left(z_{k+1} - z_k\right)\left(\frac{u_{k+1} + u_k}{2}\right)$$

$$\bar{u}_j \approx \left\{ \frac{\frac{(u_{K+1} - u_K)(z_{mc} - z_K)}{(z_{K+1} - z_K)} + 2\,u_K}{2} \right\} (z_{mc} - z_K)$$

K = index of the last radiosonde measurement point below the height of the cloud centroid at stabilization z_{mc}

and u_k are wind speeds at the radiosonde observation heights z_k. The time t_{ck} required for the centroid of the ground cloud to reach an altitude z_k for continuous sources is given by the expression

$$t_{ck} = s^{-1/2} \arccos\left(1 - \frac{(z_k)^3}{J}\right); \left((z_k)^3 / J\right) > 2 \qquad (2\text{-}9)$$

where

$$J = \frac{3\,Q_c\ z_k}{\rho\ c_p\ \pi \gamma_c^2\ \frac{\Delta \Phi}{\Delta z} \sum_{i=1}^{k} (\bar{u}_i)}$$

The time t^* required for the cloud to reach the stabilization height z_{mc} is identical to the time for instantaneous sources given by Equation (2-4).

As in the case of instantaneous sources, iteration is required to solve for the height z_{mc} because it is interrelated with the height over which $\Delta\Phi/\Delta z$ is measured and with the wind speed $\bar{u}_c$. The same iteration procedure described in Section 2.2.1 above for instantaneous sources is used in the Preprocessor routine for calculating z_{mc}.

2.3 CALCULATION OF SOURCE MODEL INPUT PARAMETERS

The Preprocessor Program is used to calculate the dimensions and spatial position of the stabilized ground cloud and the distribution of exhaust products within the cloud. In its present form, the Preprocessor Program can be used only with Models 3 and 4 (see Section 3) to calculate the source model input parameters for those portions of the stabilized ground cloud that are contained within the surface mixing layer. The source geometry identified with Models 3 and 4 is described in Section 5. In brief, Model 3 assumes that the portion of the stabilized ground cloud in the surface mixing layer is all contained within a spherical volume source. In Model 4, the stabilized ground cloud in the surface mixing layer is assumed to be contained in a number of cylindrical volume sources extending from the ground surface to the top of the mixing layer. Extension of the Preprocessor Program to include calculations of source inputs for all the models in the NASA/MSFC Multi-layer Diffusion Models Program is currently in process with emphasis on implementation for the REEDA system of NASA/MSFC.

2.3.1 Calculation of the Dimensions and Spatial Position of the Stabilized Ground Cloud

In the Preprocessor Program, the cloud radius at any height z during cloud rise is calculated by the expression

$$r\{z\} = \begin{cases} \gamma z & ; \quad z \leq z_m \\ \gamma(2z_m - z) ; & z_m < z \leq 2z_m \\ 0 & ; \quad z > 2z_m \end{cases} \tag{2-10}$$

where γ is equal γ_I or γ_c and z_m is equal to z_{mI} or z_{mc} depending on whether the source is instantaneous (I) or continuous (c).

The alongwind, crosswind and vertical source dimensions in the surface mixing layer for Model 3 are calculated under the following assumptions:

- The alongwind, crosswind and vertical distributions of exhaust products within the stabilized cloud are Gaussian
- The concentration of exhaust products at a distance of one radius from the cloud centroid is 10 percent of the concentration at the centroid

Thus the standard deviations of the alongwind ($\sigma_{xo}\{K = 1\}$), crosswind ($\sigma_{yo}\{K= 1\}$) and vertical ($\sigma_{zo}\{K=1\}$) distribution which define the source dimensions at the point of cloud stabilization are calculated from the relationships

$$\sigma_{xo}\{K = 1\} = \sigma_{yo}\{K = 1\} = r\{z_m\} / 2.15 = \gamma z_m / 2.15 \tag{2-11}$$

$$\sigma_{zo}\{K=1\} = \begin{cases} r\{z_m\} / 2.15 & ; \; z_{TK}\{K=1\} > z_m + r\{z_m\} \\ \dfrac{z_{TK}\{K = 1\} - z_m + r\{z_m\}}{4.3} & ; \; z_m - r\{z_m\} < z_{TK}\{K=1\} \leq z_m + r\{z_m\} \end{cases} \tag{2-12}$$

where

$z_{TK}\{K = 1\}$ = depth of the surface mixing layer for Model 3

In the special case where the bulk of the cloud is above the surface mixing layer ($z_{tK}\{K = 1\} \le z_m - r\{z_m\}$), the Preprocessor Program prints a message indicating that calculations are terminated for Model 3 since its use is invalid in the surface mixing layer for this case. The effective height $H_K\{K = 1\}$ of the stabilized cloud centroid in the surface mixing layer for Model 3 is given by

$$H_K\{K=1\} = \begin{cases} z_m & ; z_{TK}\{K=1\} \ge z_m + r\{z_m\} \\ \dfrac{z_{TK}\{K=1\} + z_m - r\{z_m\}}{2} & ; z_m - r\{z_m\} < z_{TK}\{K=1\} < z_m + r\{z_m\} \end{cases} \tag{2-13}$$

In the case of Model 4, the Preprocessor Program assumes that the surface mixing layer, for which the depth is defined as $z_{TL}\{L = 1\}$, is comprised of one or more sublayers K with boundaries at heights in the surface mixing layer specified in the radiosonde message. Alongwind and crosswind source dimensions are defined in each K sublayer for normal launches by

$$\sigma_{xo}\{K\} = \sigma_{yo}\{K\} = \begin{cases} \dfrac{\gamma z_m}{2.15} & ; z' \le z_m \\ \dfrac{\gamma(2z_m - z')}{2.15} & ; z' > z_m, \gamma z' > 200 \\ \dfrac{200}{2.15} = 93 & ; z' > z_m, \gamma z' \le 200 \end{cases} \tag{2-14}$$

and for abnormal launches by

$$
\sigma_{xo}\{K\} = \sigma_{yo}\{K\} = \begin{cases} \dfrac{\gamma z_m}{2.15} & ; \quad z' \leq z_m \\[2ex] \dfrac{\gamma\,(2\,z_m - z')}{2.15} & ; \quad z_m < z' \leq z_m \\[2ex] 0 & ; \quad z' > 2\,z_m \end{cases}
$$

z' = midpoint of the K^{th} sublayer

= $(z_{TK} + z_{BK}) \,/\, 2$

z_{TK} = height of the top of the K^{th} sublayer

z_{BK} = height of the base of the K^{th} sublayer

The corresponding vertical source dimension for Model 4 is zero since the model assumes a rectangular source distribution in the sublayers and uses error functions to simulate a vertical line source between sublayer boundaries which is comprised of an infinite number of (vertical) point sources (see Section 3.4 below).

For Model 4, the spatial position in the horizontal plane of the cloud in any sublayer K at t^*, the time of cloud stabilization, with respect to the origin at the launch pad is assumed to be given by

$$A_K = 90 - \tan^{-1}(y_k / x_k) \tag{2-15}$$

$$R_K = (x_k^{\,2} + y_k^{\,2})^{1/2} \tag{2-16}$$

where

$$y_k = y_{k-1} - R_k \cos(\bar{\theta}) \tag{2-17}$$

$$x_k = x_{k-1} - R_k \sin(\bar{\theta}) \tag{2-18}$$

For $t_{I,\,k+1}$ or $t_{c,\,k+1} < t^*$: (2-19)

$$\bar{\theta} = (\theta_{k+1} + \theta_k)/2$$

$$R_k = \begin{cases} (t_{I,\,k+1} - t_{I,\,k})\,(u_{k+1} + u_k)/2 \text{ ; instantaneous source} \\[2ex] \dfrac{(t_{c,\,k+1} - t_{c,\,k}) \displaystyle\sum_{i=1}^{k} (z_{i+1} - z_i)\,(u_{i+1} + u_i)/2}{\displaystyle\sum_{i=1}^{k} (z_{i+1} - z_i)} \text{ ; continuous source} \end{cases} \tag{2-20}$$

where θ_k and u_k are respectively the wind direction and wind speed from radiosonde measurements at the k^{th} height and $t_{I,k}$ and $t_{c,k}$ are the times respectively from Equations (2-3) and (2-9) for the cloud to pass through the k^{th} height. When $t_{I,\,k+1}$ or $t_{c,\,k+1} > t^*$:

$$\bar{\theta} = (\theta_m + \theta_k)/2 \tag{2-21}$$

where

$$\theta_m = \left[\theta' / (z_{k+1} - z_k)\right]\left[z_m - z_k\right] + \theta_k \tag{2-22}$$

$$\theta' = \begin{cases} (\theta_{k+1} - \theta_k) & ; \ |\theta_{k+1} - \theta_k| < 180 \\ (\theta_{k+1} - \theta_k - 360); & \theta_{k+1} - \theta_k > 180 \\ (\theta_{k+1} - \theta_k + 360); & \theta_{k+1} - \theta_k < -180 \end{cases} \tag{2-23}$$

and

$$\mathcal{R}_k = \left\{ \begin{array}{ll} R_I' \ (t^* - t_{I,k+1}) \ ; & \text{instantaneous source} \\ R_c' \ (t^* - t_{c,k+1}) \ ; & \text{continuous source} \end{array} \right\} \tag{2-24}$$

where

$$R_I' = \left[(z_m - z_k) / (z_{k+1} - z_k)\right] \left[(u_{k+1} - u_k) / 2\right] + u_k \tag{2-25}$$

$$R_c' = \left\{ \left\{ \left[(z_m - z_k) / (z_{k+1} - z_k)\right] \left[(u_{k+1} - u_k) / 2\right] + u_k \right\} \left\{ z_m - z_k \right\} + \sum_{i=1}^{k} \left[(z_{i+1} - z_i)(u_{i+1} + u_i) / 2\right] \right\} / \left\{ z_m \right\} \tag{2-26}$$

For Model 3, only the last position of the stabilized cloud, obtained by stepping through Equations (2-15) to (2-26), is used to identify the position of the cloud at stabilization.

2.3.2 Calculation of the Distribution of Exhaust Products within the Stabilized Cloud

The fraction by weight of pollutant material in the stabilized ground cloud in each of the K sublayers within the surface mixing layer for Model 4 is given by the expression

$$F\{K\} = \left\{ \begin{array}{ll} Q\,P\,\{z_{TK}\} & ; \ K = 1 \\ Q\,(P\,\{z_{TK}\} - P\,\{z_{BK}\}); & 1 < K < z_{TL}\,\{L = 1\} \end{array} \right\} \tag{2-27}$$

where

Q = total weight of exhaust products in the stabilized ground cloud

$$= \quad W \cdot t_R \{z_m\} \cdot FM \tag{2-28}$$

FM = fraction by weight of pollutant material in the fuel from Table 2-2

$$P\{z_{TK}\} = \frac{1}{\sqrt{2\pi}\ \sigma} \int_{-\infty}^{z_{TK}} \exp\left[-1/2\left(\frac{z - z_m}{\sigma}\right)^2\right] dz \tag{2-29}$$

$$= \Phi\left\{\frac{z_{TK} - z_m}{\sigma}\right\}$$

$$\sigma = r\{z = z_m\} / 2.15$$

$$P\{z_{BK}\} = \frac{1}{\sqrt{2\pi}\ \sigma} \int_{-\infty}^{z_{BK}} \exp\left[-1/2\left(\frac{z - z_m}{\sigma}\right)^2\right] dz \tag{2-30}$$

$$= \Phi\left\{\frac{z_{BK} - z_m}{\sigma}\right\}$$

$$\Phi(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{t} \exp\left(\frac{-\xi^2}{2}\right) d\xi$$

Inspection of Equations (2-29) and (2-30) shows that a Gaussian vertical distribution of material is assumed about the height of the stabilized ground cloud z_m. If the height z_m is less than the depth of the surface mixing layer $z_{TL}\{L = 1\}$ and the launch is normal, the vehicle exhaust trail will insert more material into the surface mixing layer than given by Equation (2-27). In this case, the total fraction of material in the K^{th} sublayer is given by

$$F_T\{K\} = \begin{cases} F\{K\} + W \cdot FM\left(t_R\{z_{TK}\} - t_R\{z_m\}\right); \; t_R\{z_{BK}\} < t_R\{z_m\}, \; z_m < z_n \; \{L=1\} \\ \\ F\{K\} + W \cdot FM\left(t_R\{z_{TK}\} - t_R\{z_{BK}\}\right); \; t_R\{z_{BK}\} > t_R\{z_m\}, \; z_m < z_{TL} \; \{L=1\} \end{cases} \tag{2-31}$$

In case of Model 3, only the total source strength of the pollutant in the surface mixing layer $F_T\ \{K = 1\}$ is required. Thus,

$$F_T\{K = 1\} = Q\ P\ \{z_{TK}\ \{K = 1\}\} \tag{2-32}$$

where $z_{TK}\{K=1\}$ is the top of the surface mixing layer, Q is defined by Equation (2-28) above and $P\{z_{TK}\{K=1\}\}$ is defined by Equation (2-29).

Since the desired concentration units for HCl, CO and CO_2 are parts per million, the complete expression for Q_K is

$$Q_K = F_T\{K\}\left(\frac{10^3\ mg}{g}\right)\left(\frac{22.4}{M}\right)\left(\frac{T}{273.16}\right)\left(\frac{1013.2}{P}\right) \tag{2-33}$$

where

M	=	molecular weight of HCl, CO, or CO_2
T	=	ambient air temperature (^oK)
P	=	ambient pressure (mb)

For Al_2O_3, the desired concentration units are milligrams per cubic meter, so that

$$Q_K = F_T\{K\}(10^3 \text{ mg/g}) \tag{2-34}$$

2.4 CALCULATION OF METEOROLOGICAL MODEL INPUT PARAMETERS

The majority of meteorological model input parameters required by the Preprocessor Program, and by Models 3 and 4 described in Section 3 below are obtained from radiosonde observations. These include:

u_k = wind speed (m sec^{-1})

θ_k = wind direction (deg)

T_k = ambient air temperature (^oK)

$(RH)_k$ = ambient relative humidity expressed as a percentage

P_k = atmospheric pressure (mb)

ρ = surface air density (gm^{-3})

In addition, the use of Models 3 and 4 to predict concentrations and dosages in the surface mixing layer requires the following meteorological parameters:

$\sigma'_{AL}\{L=1\}$ or $\sigma'_{AK}\{K=1\}$ = mean layer standard deviation of the wind azimuth angle in radians for the surface mixing layer for Model 4 (L notation) and Model 3 (K notation).

$\sigma'_{EL}\{L=1\}$ or $\sigma'_{EK}\{K=1\}$ = mean layer standard deviation of the wind elevation angle in radians for the surface mixing layer for Model 4 and Model 3.

α_L or α_K = lateral diffusion coefficient in the surface mixing layer for Model 4 and Model 3.

β_L or β_K = vertical diffusion coefficient in the surface mixing layer for Model 4 and Model 3.

$z_{TL}\{L=1\}$ or $z_{TK}\{K=1\}$ = depth of the surface mixing layer for Model 4 and Model 3.

The standard deviation of the wind azimuth angle in the surface mixing layer, in the absence of turbulence induced in the layer by the rocket vehicle itself, is proportional to the source function time τ and the wind speed profile in the mixing layer. Dumbauld, et al. (1970) give the following relationships:

$$\sigma'_A\{\tau\} \sim \sigma'_{AR}\left(\frac{\tau}{\tau_o}\right)^{1/5} \; ; \; \tau < \tau_o < 600 \text{ seconds} \tag{2-35}$$

and

$$\sigma'_A\{z\} \sim \sigma'_{AR}\left(\frac{z}{z_R}\right)^{-p} \tag{2-36}$$

where

σ'_{AR} = standard deviation of the wind aximuth angle for the reference time τ_o measured at a reference height z_R

p = power-law coefficient of the wind speed profile in the surface mixing layer

Assuming that the source function time τ is equivalent to the time t^* required for stabilization of the ground cloud, the mean standard deviation of the wind azimuth angle in the surface mixing layer is

$$\sigma'_{AL}\{L=1\} = \sigma'_{AR}\left(\frac{t^*}{\tau_o}\right)^{1/5} \frac{\left(z_{TL}\{L=1\}^{1-p} - z_R^{1-p}\right)}{\left(z_{TL}\{L=1\} - z_R\right)(1-p)(z_R)^{-p}} \qquad (2\text{-}37)$$

For physically reasonable combinations of z_{TL}, p and t^*, which are interrelated because of their joint dependence on atmospheric stability in the surface mixing layer, the value of $\sigma'_{AL}\{L = 1\}$ falls within the range

$$\frac{\sigma'_{AR}\{\tau_o = 600\}}{4} < \sigma'_{AL}\{L = 1\} < \frac{\sigma'_{AR}\{\tau_o = 600\}}{1.5} \qquad (2\text{-}38)$$

when the reference standard deviation σ'_{AR} is measured near the surface ($2\text{ m} \le z_R \lesssim 20\text{ m}$) over a reference time of 10 minutes. The passage of the rocket vehicle through the surface layer introduces trubulence that may persist for 30 minutes or longer, depending on the stability. This vehicle-generated turbulence enhances ambient turbulence levels and thus acts to increase the value of $\sigma'_{AL}\{L=1\}$. For these reasons, the simple expression

$$\sigma'_{AL}\{L = 1\} = \frac{\sigma'_{AR}\{\tau_o = 600\}}{2} \qquad (2\text{-}39)$$

is used in the Preprocessor Program to define the mean layer standard deviation of the wind azimuth angle.

The standard deviation of the wind elevation angle is not related to the source function time, but is approximately related to the value of $\sigma'_A\{\tau_o\}$ by the expression (Cramer, et al., 1964)

$$\sigma'_{ER} \approx \frac{\sigma'_{AR}\{\tau_o = 600 \text{ seconds}\}}{3} \qquad (2\text{-}40)$$

and to the wind profile in the surface mixing layer by

$$\sigma'_E\{z\} \cong \begin{cases} \sigma'_{ER}\left(\dfrac{z}{z_R}\right)^{0.3-p} & ;\ \text{unstable conditions} \\[2ex] \sigma'_{ER}\left(\dfrac{z}{z_R}\right)^{-p} & ;\ \text{neutral and stable conditions} \end{cases} \qquad (2\text{-}41)$$

Therefore, under unstable atmospheric conditions, σ'_E increases with height. Also, while σ'_E decreases with height under neutral and stable conditions, vehicle-induced turbulence will persist longer under these conditions and will tend to increase ambient turbulent levels. For these reasons, the Preprocessor Program assumes turbulence is isentropic in the surface mixing layer such that

$$\sigma'_{EL}\{L=1\} = \sigma'_{AL}\{L=1\} = \frac{\sigma'_{AR}\{\tau_o = 600\}}{2} \qquad (2\text{-}42)$$

Thus, the only turbulence input required for program operation is $\sigma'_{AR}\{\tau_o = 600$ seconds$\}$, the reference standard deviation of the wind azimuth angle measured over a 10-minute period at a height near the surface. At Kennedy Space Center, these measurements are available from bi-directional vanes mounted on towers throughout the launch complex areas.

In the NASA/MSFC Multilayer Diffusion Models Program, values must be assigned to the lateral α and vertical β power-law diffusion coefficients. Experience in application of the models to predict dispersion from nearly-instantaneous sources has shown that α and β normally vary insignificantly from unity. The diffusion experiments discussed by Pasquill (1974) indicate that α and β for an instantaneous source are essentially independent of atmospheric stability. For short travel distances, the experimental evidence suggests that α is greater than or equal to unity. Even for travel distances as long as 500 kilometers, a value of unity (Pasquill 1974, p. 224) appears to provide a reasonably good approximation to the data. Before vertical cloud expansion is restricted at the top of the surface mixing layer, the diffusion experiments analyzed by Pasquill show that β is also equal to unity. At larger downwind distances, the experiments indicate a less rapid increase of cloud vertical expansion, which probably reflects the restriction on vertical growth at the top of the surface mixing layer. Since the NASA/MSFC Multilayer Diffusion Models provide for restricting vertical growth at the top of the surface mixing layer, a value for β of unity would provide a good estimate of vertical expansion rates. For these reasons, the Preprocessor Program sets all power-law diffusion coefficients in the surface mixing layer to unity.

The depth of the surface mixing layer, $z_{TL}\{L = 1\}$ for Model 4 and $z_{TK}\{K = 1\}$ for Model 3, is not automatically derived from meteorological data available to the Preprocessor Program at the present time, but must be specified as an input to the program. In the input list to the Preprocessor Program, the depth of the surface mixing layer is denoted by H_m. An experienced meteorologist can estimate the depth of the surface mixing layer through inspection of the radiosonde data and a general knowledge of the mesoscale and synoptic scale weather patterns at a particular site.

SECTION 3

THE NASA/MSFC MULTILAYER DIFFUSION MODEL

The meteorological structure in the lower troposphere is usually comprised of several layers with distinctive wind, temperature and humidity fields. Horizontal spatial variations in wind regimes can also occur in the surface layer, usually as a consequence of changes in terrain or at land-water interfaces. The models described below have been designed to accommodate to these variations of meteorological structure in the lower troposphere. The vertical stratification problem in the troposphere is handled by applying the models to individual layers in which the meteorological structure is reasonably homogeneous. Layer boundaries are placed at the points of major discontinuities in the vertical profiles of wind, temperature, and humidity. For simplicity, it is assumed that, in general, there is no flux of material across layer boundaries due to turbulent mixing. Provision is made, however, for the flux of material across layer boundaries as a result of gravitational settling or precipitation scavenging and, in Model 4, as a result of breakdown in meteorological layer structure. Changes in the meteorological structure of layers, at some arbitrary time or downwind distance from the point of release, are accommodated by stopping the transport and diffusion processes in the layers affected by the change in structure, calculating new sets of initial source and meteorological input parameters and re-starting the transport and diffusion process with the new inputs. The provisions in the use of Model 4 for permitting changes in initial conditions are also ideally suited for calculating concentration and dosage fields in the surface mixing layer when portions of the cloud of exhaust products are located at varying spatial positions at cloud stabilization. The use of this feature of Model 4 is further explained in Section 3.4 below.

The major changes in the model construct from those appearing in the report by Dumbauld, Bjorklund and Bowers (1973) are that provision has been made in the Vertical Terms of Models 3, 4 and 6 for partial reflection of material at the ground or water surface. For convenience, the mathematical specifications for all the various layer models contained in the NASA/MSFC Multilayer Diffusion Models Program are given below.

3.1 MODEL 1

In this layer model, the source extends vertically through the entire layer and turbulent mixing is occurring. It is assumed that the vertical distribution of toxic material is uniform with height and that the distributions of toxic material along the x- and y-layer coordinates are Gaussian.

3.1.1 Dosage Equation for Model 1

The dosage equation for Model 1 in the K^{th} layer is

$$D_K\left\{x_K, y_K, z_K\right\} = \frac{Q_K}{\sqrt{2\pi}\,\bar{u}_K\,\sigma_{yK}\left(z_{TK} - z_{BK}\right)}\left\{\exp\left(\frac{-y_K^2}{2\sigma_{yK}^2}\right)\right\} \tag{3-1}$$

In the above expression

Q_K = the source strength in units of mass

z_{TK} = height of the top of the K^{th} layer

z_{BK} = height of the base of the K^{th} layer

The quantity $\bar{u}_K$ in Equation (3-1) is the mean cloud transport speed in meters per second in the K^{th} layer. In the surface layer ($K = 1$), the wind speed-height profile is defined according to the power-law expression

$$\bar{u}\left\{z_K, K = 1\right\} = \bar{u}_R \left(\frac{z_K \{K = 1\}}{z_R}\right)^p \tag{3-2}$$

where

$\bar{u}_R$ = mean wind speed measured at the reference height z_R

p = power-law exponent for the wind speed profile in the surface layer

$= \log\left(\frac{\bar{u}_{TK}\{K=1\}}{\bar{u}_R}\right) \Big/ \log\left(\frac{z_{TK}\{K=1\}}{z_R}\right)$

$\bar{u}_{TK}\{K=1\}$ = mean wind speed at the top of the surface layer $z_{TK}\{K=1\}$

$z_K\{K=1\}$ = height in the surface layer

Thus, in the surface layer, the mean cloud transport speed is defined by the expression

$$\bar{u}_K\{K=1\} = \frac{\bar{u}_R}{\left(z_{TK}\{K=1\} - z_R\right) z_R^{\,p}} \int_{z_R}^{z_{TK}} \left(z_K\{K=1\}\right)^p dz$$

$$= \frac{\bar{u}_R\left[\left(z_{TK}\{K=1\}\right)^{1+p} - \left(z_R\right)^{1+p}\right]}{\left(z_{TK}\{K=1\} - z_R\right)\left(z_R\right)^p\left(1+p\right)}$$

In layers above the surface layer ($K > 1$), the wind speed-height profile is assumed linear and defined by the expression

$$\bar{u}\left\{z_K, K > 1\right\} = \bar{u}_{BK} + \left(\frac{\bar{u}_{TK} - \bar{u}_{BK}}{z_{TK} - z_{BK}}\right)\left(z_K - z_{BK}\right) \tag{3-3}$$

where

$\bar{u}_{TK}$ = mean wind speed at the top of the layer z_{TK}

$\bar{u}_{BK}$ = mean wind speed at the base of the layer z_{BK}

In the K^{th} layer ($K > 1$), the mean cloud transport speed is given by the expression

$$\bar{u}_K \{K > 1\} = \left(\bar{u}_{TK} + \bar{u}_{BK}\right) / 2$$

The standard deviation of the crosswind dosage distribution is defined by the expression

$$\sigma_{yK} = \left\{ \left[\sigma'_{AK} \{\tau\} \, x_{ry} \left(\frac{x_K + x_{yK} - x_{ry} \left(1 - \alpha_K\right)}{\alpha_K \, x_{ry}} \right)^{\alpha_K} \right]^2 + \left[\frac{\Delta\theta'_K \, x_K}{4.3} \right]^2 \right\}^{1/2} \tag{3-4}$$

where

$\sigma'_{AK} \{\tau\}$ = mean layer standard deviation of the wind azimuth angle in radians for the cloud stabilization time τ

In the surface layer ($K = 1$),

$$\sigma'_{AK} \left\{\tau, K=1\right\} = \frac{\sigma'_{AR} \{\tau\} \left[\left(z_{TK} \{K=1\}\right)^{m+1} - \left(z_R\right)^{m+1} \right]}{\left(m+1\right)\left(z_{TK} \{K=1\} - z_R\right) \left(z_R\right)^m} \tag{3-5}$$

where

$\sigma'_{AR}\{\tau\}$ = standard deviation of the wind azimuth angle in radians at height z_R and for the cloud stabilization time τ

$$= \sigma_{AR}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{AR}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at height z_R and for the reference time period τ_o

m = power-law exponent for the vertical profile of the standard deviation of the wind azimuth angle in the surface layer

$$= \log\left(\frac{\sigma'_{ATK}\{\tau, K=1\}}{\sigma'_{AR}\{\tau\}}\right) \Big/ \log\left(\frac{z_{TK}\{K=1\}}{z_R}\right)$$

$$\sigma'_{ATK}\{\tau, K=1\} = \sigma_{ATK}\{\tau_o, K=1\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ATK}\{\tau_o, K=1\}$ = standard deviation of the wind azimuth angle in degrees at the top of the surface layer z_{TK} for the reference time period τ_o

For layers above the surface ($K > 1$),

$$\sigma'_{ATK}\{\tau, K>1\} = \left(\sigma'_{ABK}\{\tau\} + \sigma'_{ABK}\{\tau\}\right)\Big/2 \qquad (3\text{-}6)$$

where

$$\sigma'_{ATK}\{\tau\} = \sigma_{ATK}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ATK}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at the top of the layer for the reference time period τ_o

$$\sigma'_{ABK}\{\tau\} = \sigma_{ABK}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ABK}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at the base of the layer for the reference time period τ_o

x_K = downwind distance from the source

y_K = crosswind distance from the axis of the cloud

x_{yK} = crosswind virtual distance

$$= \frac{\sigma_{yo}\{K\}}{\sigma_{AK}\{\tau\}} - x_{Ry}$$

when $\sigma_{yo}\{K\} \leq \sigma'_{AK}\{\tau\}\, x_{ry}$

$$= \alpha_K\, x_{ry}\left(\frac{\sigma_{yo}\{K\}}{\sigma'_{AK}\{\tau\}\, x_{ry}}\right)^{1/\alpha_K} - x_{Ry} + x_{ry}\,(1-\alpha_K)$$

when $\sigma_{yo}\{K\} > \sigma'_{AK}\{\tau\}\, x_{ry}$

$\sigma_{yo}\{K\}$	=	standard deviation of the lateral source dimension in the layer at downwind distance x_{Ry}
x_{Ry}	=	distance from the source at which $\sigma_{yo}\{K\}$ is measured
x_{ry}	=	distance over which rectilinear crosswind expansion occurs downwind from an ideal point source
α_K	=	lateral diffusion coefficient in the layer
$\Delta\theta'_K$	=	vertical wind direction shear in the layer
	=	$\left(\theta_{TK} - \theta_{BK}\right)\left(\frac{\pi}{180}\right)$
θ_{TK}	=	mean wind direction in degrees at the top of the layer
θ_{BK}	=	mean wind direction in degrees at the base of the layer

3.1.2 Concentration Equation for Model 1

The maximum concentration for Model 1 in the K^{th} layer is given by the expression

$$\chi_K\left\{x_K, y_K, z_K\right\} = \frac{D_K \bar{u}_K}{\sqrt{2\pi}\,\sigma_{xK}} \tag{3-7}$$

where

σ_{xK}	=	standard deviation of the alongwind concentration distribution in the layer

$$= \left[\left(\frac{L\{x_K\}}{4.3} \right)^2 + \sigma_{xo}^{\ 2}\{K\} \right]^{1/2} \tag{3-8}$$

$L\{x_K\}$ = alongwind cloud length for a point source in the layer at the distance x_K from the source

$$= \begin{cases} \dfrac{0.28\,(\Delta \bar{u}_K)\,(x_K)}{\bar{u}_K} & ;\ \Delta \bar{u}_K \geq 0 \\[2ex] \dfrac{0.28\,(|\Delta \bar{u}_K|)\,(x_K)}{\bar{u}_K} & ;\ \Delta \bar{u}_K < 0,\ \dfrac{\Delta \Phi}{\Delta z}\{K\} < 0 \\[2ex] 0 & ;\ \Delta \bar{u}_K < 0,\ \dfrac{\Delta \Phi}{\Delta z}\{K\} \geq 0 \end{cases} \tag{3-9}$$

$\Delta \bar{u}_K$ = vertical wind-speed shear in the layer

$\Delta \bar{u}_K\{K=1\}$ = $\bar{u}_{TK}\{K=1\} - \bar{u}_R$

$\Delta \bar{u}_K\{K>1\}$ = $\bar{u}_{TK} - \bar{u}_{BK}$

$\dfrac{\Delta \Phi}{\Delta z}\{K\}$ = lapse rate of virtual potential temperature in the layer

$\sigma_{xo}\{K\}$ = standard deviation of the alongwind source dimension in the layer at the point of cloud stabilization

The above equation for $L\{x_K\}$ is based on the theoretical and empirical results reported by Tyldesley and Wallington (1965) who analyzed ground-level concentration measurements made at distances of 5 to 120 kilometers downwind from instantaneous line-source releases.

The maximum centerline concentration for Model 1 in the K^{th} layer is given by the expression

$$\chi_{CK}\left\{x_K, y_K = 0, z_K\right\} = \chi_K \left\{\exp\left(\frac{-y_K^{\ 2}}{2\,\sigma_{yK}^{\ 2}}\right)\right\}^{-1} \tag{3-10}$$

The average alongwind concentration is defined as

$$\bar{\chi}_K = D_K / t_{pK} \tag{3-11}$$

where

t_{pK} = cloud passage time in seconds in the K^{th} layer

$\cong 4.3\ \sigma_{xK} / \bar{u}_K$

The time mean alongwind concentration in the K^{th} layer is defined by the expression

$$\chi_K\left\{x_K, y_K, z_K; T_A\right\} = \frac{D_K}{T_A}\left\{\mathrm{erf}\left(\frac{\bar{u}_K\,T_A}{2\sqrt{2}\,\sigma_{xK}}\right)\right\} \tag{3-12}$$

where

T_A = time in seconds over which concentration is to be averaged

The time mean alongwind concentration is equivalent to the average alongwind concentration when t_{pK} equals T_A.

3.2 MODEL 2

Layer Model 2 refers to the same source configuration as Model 1 in which the source extends vertically through the entire depth of the layer and the distribution of toxic material is uniform with height. In Model 2, however, it is assumed that no turbulent mixing is occurring. Consequently, there is no dilution of the cloud due to turbulent expansion. The dosage and concentration equations for Model 2 are given by Equations (3-1) and (3-7), respectively, with the following substitutions:

$$\sigma_{yK} = \sigma_{yo}\{K\} \tag{3-13}$$

$$\sigma_{xK} = \sigma_{xo}\{K\} \tag{3-14}$$

3.3 MODEL 3

This layer model differs from Models 1 and 2 in that the vertical extent of the source is less than the depth of the layer. The model equation thus contains vertical expansion terms.

3.3.1 Dosage Equation for Model 3

The dosage equation for Model 3 in the K^{th} layer is given by the expression

$$D_K\left\{x_K, y_K, z_{BK} < z_K < z_{TK}\right\} = \frac{Q_K}{2\pi\,\sigma_{yK}\,\sigma_{zK}\,\bar{u}_K}\left\{\exp\left[-\frac{1}{2}\left(\frac{y_K}{\sigma_{yK}}\right)^2\right]\right\} \tag{3-15}$$

$$\left\{\sum_{i=0}^{\infty}\left[\gamma_r^i\left[\exp\left(-\frac{1}{2}\left(\frac{2i\left(z_{TK}-z_{BK}\right)+\left(H_K-z_K\right)}{\sigma_{zK}}\right)^2\right]\right]\right.$$

$$+ \gamma_r^{i+1} \left[\exp \left(- \frac{1}{2} \left(\frac{2i\left(z_{TK} - z_{BK}\right) + \left(H_K - 2z_{BK} + z_K\right)}{\sigma_{zK}} \right)^2 \right) \right] \Bigg]$$

$$+ \sum_{i=1}^{\infty} \left[\gamma_r^i \left[\exp \left(- \frac{1}{2} \left(\frac{2i\left(z_{TK} - z_{BK}\right) - \left(H_K - z_K\right)}{\sigma_{zK}} \right)^2 \right) \right] \right. \tag{3-15}$$

$$\left. + \gamma_r^{i-1} \left[\exp \left(- \frac{1}{2} \left(\frac{2i\left(z_{TK} - z_{BK}\right) - \left(H_K - 2z_{BK} + z_K\right)}{\sigma_{zK}} \right)^2 \right) \right] \right] \Bigg\}$$

where

Q_K	=	source strength or total mass of material in the layer
H_K	=	effective source height or height of the centroid of the stabilized cloud
σ_{zK}	=	standard deviation of the vertical dosage distribution in the layer
γ_r	=	fraction of material reflected at the surface z_{BK} (1 for complete reflection and 0 for no reflection) and 0^0 is defined to be equal to unity for convenience in writing Equation (3-15)

The remaining terms are the same as those in Equation (3-1) for Model 1.

The standard deviation of the vertical dosage distribution is defined by the expression

$$\sigma_{zK} = \sigma'_{EK} \, x_{rz} \left(\frac{x_K + x_{zK} - x_{rz} \left(1 - \beta_K\right)}{\beta_K \, x_{rz}} \right)^{\beta_K} \tag{3-16}$$

where

σ'_{EK}	=	mean standard deviation of the wind elevation angle in radians for the layer
x_{zK}	=	vertical virtual distance in the layer
β_K	=	vertical diffusion coefficient in the layer
x_{rz}	=	distance over which rectilinear vertical expansion occurs downwind from an ideal point source

In the surface layer (K = 1),

$$\sigma'_{EK}\{K=1\} \quad \frac{\sigma_{ER}\left[\left(z_{TK}\{K=1\}\right)^{q+1} - \left(z_R\right)^{q+1}\right]}{\left(q+1\right)\left(z_{TK}\{K=1\} - z_R\right)\left(z_R\right)^q} \left(\frac{\pi}{180}\right) \tag{3-17}$$

where

σ_{ER}	=	standard deviation of the wind elevation angle in degrees at the height z_R
q	=	power-law exponent for the vertical profile of the standard deviation of the wind elevation angle in the surface layer

$$= \log \left(\frac{\sigma_{ETK} \{K=1\}}{\sigma_{ER}} \right) \Big/ \log \left(\frac{z_{TK} \{K=1\}}{z_R} \right)$$

$\sigma_{ETK} \{K=1\}$ = standard deviation of the wind elevation angle in degrees at the top of the surface layer

Above the surface layer (K > 1),

$$\sigma'_{EK} \{K > 1\} = \left(\sigma_{ETK} + \sigma_{EBK} \right) \left(\frac{\pi}{360} \right)$$

where

σ_{ETK} = standard deviation of the wind elevation angle in degrees at the top of the layer

σ_{EBK} = standard deviation of the wind elevation angle in degrees at the base of the layer

The vertical virtual distance x_{zK} is given by the expression

$$\left\{ \begin{array}{ll} \dfrac{\sigma_{zo} \{K\}}{\sigma'_{EK}} - x_{Rz} & ; \ \sigma_{zo} \{K\} \leq \sigma'_{EK} \, x_{rz} \\[2ex] \beta_K \, x_{rz} \left(\dfrac{\sigma_{zo} \{K\}}{\sigma'_{EK} \, x_{rz}} \right)^{1/\beta_K} - x_{Rz} + x_{rz} \left(1 - \beta_K \right) & ; \ \sigma_{zo} \{K\} > \sigma'_{EK} \, x_{rz} \end{array} \right\}$$

where

$\sigma_{zo} \{K\}$ = standard deviation of the vertical dosage distribution at x_{Rz}

x_{Rz} = distance from the source at which $\sigma_{zo} \{K\}$ is measured in the K^{th} layer

3.3.2 Concentration Equation for Model 3

The concentration equation for Model 3 is the same as that for Model 1 which is given by Equation (3-7) in Section 3.1.2 with D_K from Equation (3-15). Equation (3-10) in Section 3.1.2 also gives the maximum centerline concentraion for Model 3. Similarly, average and time mean alongwind concentrations for Model 3 are given by Equations (3-11) and 3-12) with D_K from Equation (3-15).

3.4 MODEL 4

Model 4, the layer-breakdown model, may be used to calculate concentration and dosage fields resulting from changes in the meteorological layer structure. Model 4 may also be used to determine concentration and dosage fields in the surface mixing layer downwind from a source in which the source strength varies with height in the layer and/or where there are spatial differences in source locations within a layer. The application of Model 4 requires the following assumption:

- The boundary between adjacent layers or sublayers is eliminated and the layers are replaced by a single layer L
- Turbulent mixing is occurring in layer L
- The material in each of the layers or sublayers is initially uniformly distributed in the vertical
- Reflection occurs at the upper and lower boundaries of layer L

The selection of Model 4 for layer breakdown calculations or to accommodate vertical source strength variations and/or spatial differences in source locations in the surface mixing layer is controlled in the computer program by selection of certain options (see Appendix B) available in the input configuration. If no special provision is made and Model 4 is specified for use, the program assumes that the function of the model is to accommodate to vertical source strength variations or spatial differences in source locations. For example, the surface mixing layer can be divided into several sublayers where the source strength, although assumed to be vertically uniform in the K^{th} sublayer, increases with height in subsequent layers. Also, because of wind speed and direction shear in the sublayers, portions of the exhaust cloud may be located at different positions in the horizontal plane at the time of cloud stabilization. In this case, Model 4 calculates the contribution from each sublayer to the composite concentration and dosage fields in the surface mixing layer by permitting turbulent mixing across the initial sublayer boundaries.

If Model 4 is to be used to predict the concentration and dosage fields downwind from a change in meteorological structure, the program option ISKIP(7) must be properly set, the input parameter NBK must be initialized, and the meteorological parameters for the new L^{th} layer and the time t^* at which layer breakdown occurs must be specified (see Appendix B).

3.4.1 Dosage Equation for Model 4

The dosage equation for Model 4 for the contribution from the portion of the cloud in the K^{th} layer to the receptor position in the layer L is given by the expression

$$D_{LK} = \frac{Q_K}{2\sqrt{2\pi}\,\bar{u}_L\,\sigma_{yLK}\left(z_{TK} - z_{BK}\right)} \left\{ \exp\left[-\frac{1}{2}\left(\frac{y_L}{\sigma_{yLK}}\right)^2 \right] \right\}$$

$$\left\{ \sum_{i=0}^{\infty} \left[\gamma_r^i \left[\operatorname{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) + z_{TK} - z_L}{\sqrt{2}\,\sigma_{zLK}} \right) + \operatorname{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) - z_{BL} + z_L}{\sqrt{2}\,\sigma_{zLK}} \right) \right] \right. \right. \quad (3\text{-}18)$$

$$\left. \left. + \gamma_r^{i+1} \left[\operatorname{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) - 2z_{BL} + z_{TK} + z_L}{\sqrt{2}\,\sigma_{zLK}} \right) + \operatorname{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) + 2z_{BL} - z_{BK} - z_L}{\sqrt{2}\,\sigma_{zLK}} \right) \right] \right] \right\}$$

$$+ \sum_{i=1}^{\infty} \left[\gamma_r^i \left[\operatorname{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) - z_{BK} + z_L}{\sqrt{2}\, \sigma_{zLK}} \right) + \operatorname{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) + z_{TK} - z_L}{\sqrt{2}\, \sigma_{zLK}} \right) \right] \right.$$

$$\left. + \gamma_r^{i-1} \left[\operatorname{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) + 2z_{BL} - z_{BK} - z_L}{\sqrt{2}\, \sigma_{zLK}} \right) + \operatorname{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) - 2z_{BL} + z_{TK} + z_L}{\sqrt{2}\, \sigma_{zLK}} \right) \right] \right] \tag{3-18}$$

where, again for convenience, 0^0 is defined equal to unity.

The total contribution to a receptor position in layer L is calculated by summing the contributions from all K layers. A derivation of the vertical term (terms following the summation signs) in Equation (3-18) is presented in Appendix E. In the above expression

Q_K = source strength in units of mass for the source in the layer K

$\bar{u}_L$ = mean cloud transport speed in the L^{th} layer

$$= \frac{\sum_{K=1}^{n} \left\{ \left(z_{TK} - z_{BK}\right) \left(\bar{u}_{TK} + \bar{u}_{BK}\right) \Big/ 2 \right\}}{\sum_{K=1}^{n} \left(z_{TK} - z_{BK}\right)} \tag{3-19}$$

$\bar{u}_{TK}$ = wind speed at the top of the K^{th} layer

$\bar{u}_{BK}$ = wind speed at the base of the K^{th} layer

and n is the number of sublayers in layer L.

The crosswind distance from the axis of the cloud to a receptor y_L (defined positive to the right looking downwind) is given by the expression

$$y_L = (y_j - y_{SK}) \sin \theta'_L - (x_j - x_{SK}) \cos \theta'_L \qquad (3\text{-}20)$$

where

x_j, y_j = position of the receptor with respect to the origin of the reference coordinate system with the y axis positive northward and the x axis positive eastward

x_{SK}, y_{SK} = coordinates of the cloud centroid in the K^{th} layer at time t^* with respect to the origin of the reference coordinate system

x_{SK} = $x_i - \bar{u}_K t^* \sin \theta'_K$

y_{SK} = $y_i - \bar{u}_K t^* \cos \theta'_K$

x_i, y_i = coordinates of the source in the K^{th} layer with respect to the origin of the reference coordinate system

$$\theta'_L = \frac{\sum_{K=1}^{n} \left\{ \left(z_{TK} - z_{BK} \right) \left(\theta'_K \right) \right\}}{\sum_{K=1}^{n} \left(z_{TK} - z_{BK} \right)} \qquad (3\text{-}21)$$

$$\theta'_K = \left(\theta_{TK} + \theta_{BK} \right) \left(\frac{\pi}{360} \right)$$

θ_{TK} = wind direction at the top of the K^{th} layer in degrees

θ_{BK} = wind direction at the base of the K^{th} layer in degrees

The standard deviation of the crosswind dosage distribution σ_{yLK} in the L^{th} layer is defined by the expression

$$\sigma_{yLK} = \left\{ \left[\sigma'_{AL} \{ \tau \} x_{ry} \left(\frac{x_L + x^*_{yKL} - x_{ry} \; (1 - \alpha_L)}{\alpha_L \, x_{ry}} \right)^{\alpha_L} \right]^2 + \left[\frac{\Delta \theta'_L \, x_L}{4.3} \right]^2 \right\}^{1/2} \tag{3-22}$$

where

$\sigma'_{AL}\{ \tau \}$ = mean layer standard deviation of the wind azimuth angle in radians for the effective cloud stabilization time τ

In the surface lyer (L = 1),

$$\sigma'_{AL}\{ \tau, L=1 \} = \frac{\sigma'_{ARL}\{\tau\} \left[\left(z_{TL}\{L=1\} \right)^{m_L + 1} - (z_R)^{m_L + 1} \right]}{(m_L + 1) \left(z_{TL}\{L=1\} - z_R \right) (z_R)^{m_L}} \tag{3-23}$$

where

$\sigma'_{ARL}\{ \tau \}$ = standard deviation of the wind azimuth angle in radians at height z_R and for time τ

$$= \quad \sigma_{ARL}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ARL}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at height z_R and for the reference time period τ_o

m_L = power-law exponent for the vertical profile of the standard deviation of the wind azimuth angle in the surface layer L = 1

$$m_L = \log\left(\frac{\sigma_{ATL}\{\tau, L=1\}}{\sigma_{ARL}\{\tau\}}\right) \Big/ \log\left(\frac{z_{TL}\{L=1\}}{z_R}\right)$$

$$\sigma'_{ATL}\{\tau, L=1\} = \sigma_{ATL}\{\tau_o, L=1\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ATL}\{\tau_o, L=1\}$ = standard deviation of the wind azimuth angle in degrees at the top of the surface layer z_{TL} for the reference time period τ_o

For layers above the surface layer (L > 1),

$$\sigma'_{AL}\{\tau, L>1\} = \left(\sigma'_{ATL}\{\tau\} + \sigma'_{ABL}\{\tau\}\right)\Big/2 \qquad (3\text{-}24)$$

where

$$\sigma'_{ATL}\{\tau\} = \sigma_{ATL}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ATL}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at the top of the layer for the reference time period τ_o

$$\sigma'_{ABL}\{\tau\} = \sigma_{ABL}\{\tau_o\}\left(\frac{\tau}{\tau_o}\right)^{1/5}\left(\frac{\pi}{180}\right)$$

$\sigma_{ABL}\{\tau_o\}$ = standard deviation of the wind azimuth angle in degrees at the base of the layer for the reference time τ_o

The wind-direction shear in radians in the layer is given by the expression

$$\Delta\theta'_L = \left(\theta_{TL} - \theta_{BL}\right)\left(\frac{\pi}{180}\right)$$

where

θ_{TL} = mean wind direction in degrees at the top of the layer z_{TL}

θ_{BL} = mean wind direction in degrees at the base of the layer z_{BL}

The crosswind virtual distance in the L^{th} layer due to source (cloud) originating in the K^{th} layer is given by the expression

$$x^*_{yKL} = x_{ry} \left(\frac{\sigma^*_{yKL}}{\sigma_{AL} \{\tau\} \; x_{ry}} \right)^{1/\alpha_L} + x_{ry} \left(1 - \alpha_L\right)$$

where

σ^*_{yKL} = crosswind source dimension in Layer L due to source (cloud) originating in the K^{th} layer

$$= \left\{ \left[\left(\sigma^*_{xK} \right)^2 \sin^2 \left(\theta'_K - \theta'_L \right) \right] + \left[\left(\sigma^*_{yK} \right)^2 \cos^2 \left(\theta'_K - \theta'_L \right) \right] \right\}^{1/2}$$

σ^*_{xK} = alongwind standard deviation of the dosage distribution in the K^{th} layer at time t*

σ^*_{yK} = crosswind standard deviation of the dosage distribution in the K^{th} layer at time t*

α_L = lateral diffusion coefficient in the layer

x_{ry} = distance over which rectilinear crosswind expansion occurs downwind from an ideal point source

The downwind distance from the point where the change in layer structure occurs for the source (cloud) in the K^{th} layer to the point where the dosage is to be calculated x_L is given by the expression

$$x_L = -(x_j - x_{SK}) \sin \theta'_L - (y_j - y_{SK}) \cos \theta'_L \qquad (3\text{-}25)$$

The standard deviation of the vertical dosage distribution σ_{zLK} in the L^{th} layer is defined by the expression

$$\sigma_{zLK} = \sigma'_{EL}\, x_{rz} \left(\frac{x_L}{x_{rz}}\right)^{\beta_L} \tag{3-26}$$

where

σ'_{EL}	=	mean standard deviation of the wind elevation angle in radians for the layer
β_L	=	vertical diffusion coefficient in the layer
x_{rz}	=	distance over which rectilinear vertical expansion occurs downwind of an ideal point source

In the surface layer (L = 1),

$$\sigma'_{EL}\{L=1\} = \frac{\sigma_{ERL}\left[\left(z_{TL}\{L=1\}\right)^{q_L+1} - \left(z_R\right)^{q_L+1}\right]}{\left(q_L+1\right)\left(z_{TL}\{L=1\} - z_R\right)\left(z_R\right)^{q_L}} \left(\frac{\pi}{180}\right) \tag{3-27}$$

where

σ_{ERL}	=	standard deviation of the wind elevation angle in degrees at the reference height z_R
q_L	=	power-law exponent for the vertical profile of the standard deviation of the wind elevation angle in the surface layer
	=	$\log\left(\frac{\sigma_{ETL}\{L=1\}}{\sigma_{ERL}}\right) \Big/ \log\left(\frac{z_{TL}\{L=1\}}{z_R}\right)$
$\sigma_{ETL}\{L=1\}$	=	standard deviation of the wind elevation angle in degrees at the top of the layer z_{TL}

Above the surface layer (L > 1),

$$\sigma'_{EL}\{L > 1\} = \left(\sigma_{ETL} + \sigma_{EBL}\right)\left(\frac{\pi}{360}\right) \tag{3-28}$$

where

σ_{ETL} = standard deviation of the wind elevation angle in degrees at the top of the layer z_{TL}

σ_{EBL} = standard deviation of the wind elevation angle in degrees at the base of the layer z_{BL}

3.4.2 Concentration Equation for Model 4

The maximum concentration equation for Model 4 is given by the expression

$$\chi_{LK}\{x_L, y_L, z_L\} = \frac{D_{LK}\,\bar{u}_L}{\sqrt{2\pi}\;\sigma_{xLK}} \tag{3-29}$$

where

σ_{xLK} = standard deviation of the cloud alongwind concentration distribution in the layer

$$= \left\{\left(\frac{L\{x_{LK}\}}{4.3}\right)^2 + \left(\sigma^*_{xKL}\right)^2\right\}^{1/2}$$

$L\{x_{LK}\}$ = alongwind cloud length of a point source at distance x_L

$$= \begin{Bmatrix} \dfrac{0.28\left(\Delta\bar{u}_L\right) x_L}{\bar{u}_L} & ; \Delta\bar{u}_L \geq 0 \\ \dfrac{0.28\left(\left|\Delta\bar{u}_L\right|\right)\left(x_L\right)}{\bar{u}_L} & ; \Delta\bar{u}_L < 0 ; \dfrac{\Delta\Phi}{\Delta z}\{L\} < 0 \\ 0 & ; \Delta\bar{u}_L < 0 ; \dfrac{\Delta\Phi}{\Delta z}\{L\} \geq 0 \end{Bmatrix} \tag{3-30}$$

$\Delta\bar{u}_L$ = vertical wind speed shear in the layer

= $\bar{u}_{TL} - \bar{u}_{BL}$

$\dfrac{\Delta\Phi}{\Delta z}\{L\}$ = lapse rate of potential temperature in the layer

σ^*_{xKL} = alongwind source dimension in layer L due to source (cloud) originating in the K^{th} layer

$$= \left\{\left[\left(\sigma^*_{xK}\right)^2 \cos^2\left(\theta'_K - \theta'_L\right)\right] + \left[\left(\sigma^*_{yK}\right)^2 \sin^2\left(\theta'_K - \theta'_L\right)\right]\right\}^{1/2}$$

The maximum centerline concentration for Model 4 in the L^{th} layer is given by the expression

$$\chi_{CLK}\{x_{LK}, y_{LK}=0, z_{LK}\} = \chi_{LK} \Big/ \{\text{LATERAL TERM}\}$$

$$= \chi_{LK}\left\{\exp\left[-\left(\frac{y_L^2}{2\sigma_{yLK}^2}\right)\right]\right\}^{-1} \tag{3-31}$$

The average alongwind concentration at the cloud centerline is defined as

$$\bar{\chi}_{LK} = D_{LK}/t_{pL} \tag{3-32}$$

where

t_{pL} = cloud passage time in seconds in the L^{th} layer

= $4.3\ \sigma_{xLK} / \bar{u}_L$

The time mean alongwind concentration in the L^{th} layer for averaging time T_A is defined by the expression

$$\bar{\chi}_K \{x_{LK}, y_{LK}, z_{LK}; T_A\} = \frac{D_{LK}}{T_A} \left\{ \operatorname{erf}\left(\frac{\bar{u}_L\ T_A}{2\sqrt{2}\ \sigma_{xLK}} \right) \right\} \quad (3\text{-}33)$$

3.5 MODEL 5

This model is used to calculate the amount of material deposited on the surface by precipitation scavenging in the K^{th} layer. The ground-level deposition WD_K due to precipitation scavenging, for the case in which the vertical distribution of toxic material in the layer is uniform with height, is given by the expression

$$WD_K\{x_K, y_K, z=0\} = \frac{\Lambda Q_K}{\sqrt{2\pi}\ \sigma_{yK}\ \bar{u}_K} \left\{ \exp\left[-\frac{1}{2}\left(\frac{y_K}{\sigma_{yK}}\right)^2 \right] \right\} \quad (3\text{-}34)$$

$$\left\{ \exp\left[-\Lambda \left(\frac{x_K}{\bar{u}_K} - t_1 \right) \right] \right\}$$

where

Q_K = source strength in units of mass for the source in layer K

t_1 = time precipitation begins

Λ = percent of material removed per unit time

The principal assumptions made in deriving the above expression are:

- The rate of precipitation is steady over an area that is large compared to the horizontal dimension of the cloud of toxic material

- The precipitation originates at a level above the top of the toxic cloud so that hydrometeors pass vertically through the entire cloud

- The time duration of the precipitation is sufficiently long so that the entire alongwind length of the toxic cloud passes over the point x

Engelmann (see Slade, 1968, pp. 208-221) discusses the general problems of calculating the amount of material removed by precipitation scavenging and recommends values of the coefficient Λ that may be combined with precipitation rates to obtain estimates of total surface deposition. Other useful information may be obtained from the proceedings of the 1970 Symposium on Precipitation Scavenging (Engelmann and Slinn, 1970), from Pellett (1974) and from Knutson, et al., (1974).

When changes in layer structure occur at time t^*, the contribution to ground deposition WD_{LK} due to precipitation scavenging in the K^{th} layer is given by the expression

$$WD_{LK}\left\{x_L, y_L, z=0\right\} = \frac{\Lambda Q_K}{\sqrt{2\pi}\;\sigma_{yLK}\;\bar{u}_L}\left\{\exp\left[-\frac{1}{2}\left(\frac{y_L}{\sigma_{yLK}}\right)^2\right]\right\} \left\{\exp\left[-\Lambda\left(\frac{x_L}{\bar{u}_L} + t^* - t_1\right)\right]\right\} \tag{3-35}$$

Maximum ground-level deposition at a point $(x_L, y_L, z=0)$ assuming no previous cloud depletion due to scavenging, can be obtained by setting the second exponential term in Equation (3-35) to unity. Total ground deposition is obtained by summing the contributions from all layers through which precipitation is falling at points on the reference grid coordinate system. The height of the top of the uppermost layer through which precipitation is falling z_{lim} must be supplied as an input to the computer program.

The dosage or concentration at a point in space, assuming precipitation scavenging occurs, is obtained by multiplying the appropriate dosage or concentration equation by the exponential term in Equation (3-34) or (3-35) containing the coefficient Λ .

3.6 MODEL 6

This model is used to calculate the ground deposition due to gravitational settling. The basic source configuration is an area source of finite lateral extent and unit vertical extent. Other source configurations are treated by summing the deposition at the ground resulting from a number of basic sources arranged to simulate the desired configuration. The model is essentially a tilted plume model in which the effects of wind shear are taken into account. The axis of a particle or droplet cloud of a given settling velocity intersects the ground plane at a distance from the source and at an angle from the mean surface wind direction that are proportional to the total angular wind shear and the residence time of the settling material in the layers between the source and the ground surface. In any layer, the inclination of the cloud axis from the horizontal is given by $\tan^{-1} V_s / \bar{u}$, where V_s is the particle or droplet settling velocity and $\bar{u}$ is the mean transport wind speed in the layer. In all cases, material released in the K^{th} layer and dispersed upwards by turbulence is assumed to be reflected downwards at the interface of the K^{th} and $(K + 1)^{th}$ layers. The basic model is used to calculate the ground-level deposition pattern for a single value of the settling velocity. The total deposition pattern is obtained by summing the results for all settling velocities representative of the particle or droplet-size distribution of the released material on a reference coordinate grid system.

In the computer program, provision is made for calculating deposition from a source which fills the layer in the vertical and for a source in which the vertical extent is less than the depth of the layer. These models are described below.

3.6.1 Gravitational Deposition Model for a Source that Extends Vertically Through the Entire Layer

Ground-level deposition by gravitational settling for a source that extends vertically through the entire layer and in which the material is uniformly distributed in the vertical is calculated by summing contributions from a number of elementary sources in the K^{th} layer. Deposition at the surface for a single elementary source at height H_{nK} in the layer is given by the expression

$$DEP_{nK} \doteq \frac{f_i \, Q_K \, T_K}{2\pi \, \sigma_{ynK} \, \xi_K} \left\{ M_{nK} + N_{nK} \right\} \left\{ \exp\left[-\frac{1}{2} \left(\frac{y_s}{\sigma_{ynK}} \right)^2 \right] \right\} \tag{3-36}$$

where

f_i	=	fraction of particles or droplets with setting velocity V_s
Q_K	=	source emission rate in layer K (g sec^{-1})
T_K	=	source emission time in layer K
ξ_K	=	number of elementary sources in layer K for simulating a uniform vertical distribution

y_s = lateral distance from the deposition axis of particles or droplets with settling velocity V_s

= $R_s \sin \phi_s$

R_s = radial distance in the horizontal plane from the source to a receptor

ϕ_s = angle between the axis of the ground-level deposition pattern and the radial connecting source and receptor for settling velocity V_s

The terms M_{nK} and N_{nK} are vertical terms that include provision for reflection from the boundary between the K^{th} and $(K+1)^{th}$ layers. These terms are defined by the expressions

$$\begin{aligned} M_{nK} + N_{nK} = (1-\gamma_r) \Bigg\{ & \left[\frac{\bar{\beta}_K H_{nK} + (1-\bar{\beta}_K) V_s x_s / \bar{u}_{nK}}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right] \left[\exp\left(-\frac{1}{2} \left(\frac{H_{nK} - (V_s x_s / \bar{u}_{nK})}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right)^2 \right) \right] \\ + \sum_{a=1}^{\infty} \gamma_r^{a-1} \Bigg\{ & \left[\frac{\bar{\beta}_K (2a z_{TK} - H_{nK}) - (1-\bar{\beta}_K) V_s x_s / \bar{u}_{nK}}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right] \left[\exp\left(-\frac{1}{2} \left(\frac{2a z_{TK} - H_{nK} + (V_s x_s / \bar{u}_{nK})}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right)^2 \right) \right] \\ + \gamma_r & \left[\frac{\bar{\beta}_K (2a z_{TK} + H_{nK}) + (1-\bar{\beta}_K) V_s x_s / \bar{u}_{nK}}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right] \left[\exp\left(-\frac{1}{2} \left(\frac{2a z_{TK} + H_{nK} - (V_s x_s / \bar{u}_{nK})}{\sigma'_{EnK}(x_s)^{1+\bar{\beta}_K}} \right)^2 \right) \right] \Bigg\} \Bigg\} \end{aligned} \tag{3-37}$$

where

x_s = $R_s \cos \phi_s$

$\bar{u}_{nK}$ = mean wind transport speed in the layer between H_{nK} and the ground

$$= \frac{\left(X_{nK}^{2} + Y_{nK}^{2}\right)^{1/2}}{H_{nK}} V_s$$

$$X_{nK} = \frac{\bar{u}_{HK}}{V_s\, b_K} \left\{ \sin\left[b_K(H_{nK} - z_{BK}) + S\theta'_{K-1} \right] - \sin\left(S\theta'_{K-1}\right) \right\}$$

$$+ \sum_{i=1}^{K-1} \left\{ \frac{\bar{u}_i}{V_s\, b_i} \left[\sin\left(S\theta'_i\right) - \sin\left(S\theta'_{i-1}\right) \right] \right\}$$

$$Y_{nK} = \frac{\bar{u}_{HK}}{V_s\, b_K} \left\{ \cos\left[b_K(H_{nK} - z_{BL}) + S\theta'_{K-1} \right] - \cos\left(S\theta'_{K-1}\right) \right\}$$

$$+ \sum_{i=1}^{K-1} \left\{ \frac{-\bar{u}_i}{V_s\, b_i} \left[\cos\left(S\theta'_i\right) - \cos\left(S\theta'_{i-1}\right) \right] \right\}$$

$$S\theta'_{K-1} = \sum_{i=1}^{K-1} \Delta\theta'_i$$

$$S\theta'_K = \sum_{i=1}^{K} \Delta\theta'_i$$

$$b_K = \frac{S\theta'_K - S\theta'_{K-1}}{z_{TK} - z_{BK}}$$

The quantity $\bar{u}_{HK}$ is the mean layer wind speed between the height H_{nK} and the base of the K^{th} layer. The following expressions define the mean layer wind speeds in the surface layer (K = 1) and the layers above the surface layer (K > 1):

$$\bar{u}_{HK}\{K=1\} = \frac{\bar{u}_R\left[\left(H_{nK}\{K=1\}\right)^{1+p} - \left(z_R\right)^{1+p}\right]}{(1+p)\left(H_{nK}\{K=1\} - z_R\right)\left(z_R\right)^p} \tag{3-38}$$

$$\bar{u}_{HK}\{K>1\} = \left[\left(\frac{\bar{u}_{TK}-\bar{u}_{BK}}{z_{TK}-z_{BK}}\right)\left(\frac{H_{nK}-z_{BK}}{2}\right)\right] + \left[\frac{\bar{u}_{BK}}{2}\right] \tag{3-39}$$

The mean standard deviation of the wind elevation angle in radians in the layer between H_{nK} and the base of the K^{th} layer is given by the expressions

$$\sigma'_{EnK}\{K=1\} = \frac{\sigma_{ER}\left[\left(H_{nK}\{K=1\}\right)^{1+q} - \left(z_R\right)^{1+q}\right]}{(1+q)\left(H_{nK}\{K=1\} - z_R\right)\left(z_R\right)^q}\left(\frac{\pi}{180}\right) \tag{3-40}$$

$$\sigma'_{EnK}\{K>1\} = \frac{1}{H_{nK}}\left\{\left[\sigma'_{EnK}\{K=1\}\right] + \left[\sum_{i=2}^{K-1} \sigma'_{Ei}\left(z_{Ti} - z_{Bi}\right)\right]\right.$$
$$\left. + \frac{\pi\left(H_{nK}-z_{BK}\right)}{360}\left[\left(\frac{\sigma_{ETK}-\sigma_{EBK}}{z_{TK}-z_{BK}}\right)\left(H_{nK} - z_{BK}\right) + \sigma_{EBK}\right]\right\} \tag{3-41}$$

The vertical diffusion coefficient in the layer between H_{nK} and the base of the K^{th} layer is given by the terms

$$\bar{\beta}_K\{K=1\} = \beta_K \tag{3-42}$$

The mean lateral diffusion coefficient in the layer between H_{nK} and the surface is given by the terms

$$\left\{ \begin{array}{l} \bar{\alpha}_K\{K=1\} = \alpha_K \\ \bar{\alpha}_K\{K>1\} = \frac{1}{H_{nK}} \left\{ \left[\sum_{i=1}^{K-1} \alpha_i \left(z_{Ti} - z_{Bi} \right) \right] + \left[\alpha_K \left(H_{nK} - z_{BK} \right) \right] \right\} \end{array} \right\} \quad (3\text{-}47)$$

The number of elementary sources ξ_K required to simulate a uniformly distributed source in the vertical is given by the expression

$$\xi_K = \left(z_{TK} - z_{BK} \right) / \Delta h_K \quad (3\text{-}48)$$

where

Δh_K = vertical separation of elementary sources in the K^{th} layer

= $R_c \sigma'_{EH} \left(X_{HK}{}^2 + Y_{HK}{}^2 \right)^{1/2} \left(1 + \frac{V_s}{\bar{u}_{HK}} \right)^{1/2}$

R_c = a constant value depending on the accuracy desired in simulating a vertical line source configuration. A value of $R_c = 0.45$ yields deposition estimates that are within 10 percent of the true value

σ'_{EH} = σ'_{EnK} with $H_{nK} = \frac{1}{3} \left(z_{TK} - z_{BK} \right) + z_{BK}$

X_{HK} = X_{nK} with $H_{nK} = \frac{1}{3} \left(z_{TK} - z_{BK} \right) + z_{BK}$

Y_{NK} = Y_{nK} with $H_{nK} = \frac{1}{3} \left(z_{TK} - z_{BK} \right) + z_{BK}$

$\bar{u}_{HK}$ = $\bar{u}_{nK}$ with $H_{nK} = \frac{1}{3} \left(z_{TK} - z_{BK} \right) + z_{BK}$

The computer program for calculating gravitational deposition automatically distributes ζ_K sources in the K^{th} layer with uniform vertical spacing. The height H_{nK} in the above equations is the height above the ground of each elementary source.

The angle between the axis of the ground-level deposition pattern and the radial connecting source and receptor for settling velocity V_s is defined by the expressions

$$\phi_s = \left| \theta_1 - 180 + \Phi_s - \theta_R \right| \quad \left(0 < \theta_1 < 180\right)$$
$$\phi_s = \left| \theta_1 + 180 + \Phi_s - \theta_R \right| \quad \left(180 < \theta_1 < 360\right) \tag{3-49}$$

where

θ_1 = mean wind direction at the reference height z_R

θ_R = angle between north and a line connecting source and receptor

$\Phi_s = \tan^{-1}\left(\frac{Y_{nK}}{X_{nK}}\right)$

3.6.2 Gravitational Deposition Model for a Volume Source in the K^{th} Layer

For a volume source at height H_{SK} in the K^{th} layer, the ground-level deposition from gravitational settling is given by the expression

$$DEP_{SK} = \frac{f_i \; Q_{SK}}{2\pi \; \sigma_{ySK}} \left\{M_{SK} + N_{SK}\right\} \left\{\exp\left[-\frac{1}{2}\left(\frac{y_{SK}}{\sigma_{ySK}}\right)^2\right]\right\} \tag{3-50}$$

where the subscript SK indicates that the parameters refer to a single souce in the K^{th} layer. The subset of equations which define the SK subscripted parameters is the same as the subset defining the terms in Equation (3-36), except the following substitution is made for the term x_s appearing in Equation (3-37):

$$x_s = R_{SK} \cos \phi_{SK} + x_{zSK} \tag{3-51}$$

where

x_{zSK} = the vertical virtual distance for the volume source

$$= \left(\frac{\sigma_{zo}\{SK\}}{\sigma'_{ESK}} \right)^{1/\beta_K}$$

σ'_{ESK} = mean standard deviation of the wind elevation angle in the layer between H_{SK} and the ground

$\sigma_{zo}\{SK\}$ = vertical source dimension of the volume source

In using Equation (3-50), deposition patterns from all values of V_{SK} representative of the particle or droplet size distribution of the volume source are summed on a reference coordinate system to obtain the total deposition pattern.

SECTION 4

DESCRIPTION OF THE NASA/MSFC MULTILAYER MODEL COMPUTER PROGRAMS

The NASA/MSFC computer programs described in this technical document consist of a Preprocessor Program and the main NASA/MSFC Multilayer Diffusion Models Program - Version 5. Both programs are written in FORTRAN IV and are designed for execution on a UNIVAC 1108 computer. In addition, the automated plotting routines in Version 5 of the program are designed to provide input to the Stromberg-Carlson (SC 4020) machine at MSFC. This section briefly describes the general characteristics of the two programs.

4.1 PREPROCESSOR PROGRAM

The Preprocessor Program is designed to automatically calculate source and meteorological model inputs for use in Models 3 and 4 of the NASA/MSFC Multilayer Diffusion Models Program - Version 5. Requisite inputs to the Preprocessor Program include vertical profiles of meteorological data of the type available from radiosonde soundings, surface turbulence data available at KSC from meteorological towers, and logic information such as the type of launch and vehicle for which the calculation is being performed. The program can currently perform calculations for the following vehicles:

- Space Shuttle
- Titan III
- Delta-Thor
- Minuteman II

Calculations performed by the Preprocessor include:

- Time-height profile of the rise of hot exhaust products contained in the ground cloud

- Position in space of the stabilized ground-cloud
- Source dimensions of the stabilized ground-cloud
- Distribution of pollutant products within the stabilized ground-cloud
- Turbulence input parameters

The cloud-rise models and algorithms used in computing this information are described in Section 2. The data decks produced on option by this program include a complete card deck for each of the four pollutants HCl, CO, CO_2 and Al_2O_3 for use in Models 3 and 4 of the NASA/MSFC Multilayer Diffusion Models Program. The program also provides the option to produce input data for normal and abnormal launches of all four vehicles. A complete description of the program with user's instructions is presented in Appendix A.

4.2 NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM - VERSION V

The NASA/MSFC Multilayer Diffusion Models Program is designed to calculate the following quantities downwind from normal and abnormal launches of rocket vehicles:

- Concentration and dosage patterns
- Time - mean concentration patterns
- Average cloud concentration
- Time of cloud passage
- Ground-level deposition patterns due to gravitational settling or precipitation scavenging

Program options include the calculation of concentration, dosage and time-mean concentration patterns with partial reflection of material at the surface, with time-dependent exponential decay, and/or with depletion due to precipitation scavenging. Also, the program is capable of calculating ground-level deposition due to gravitational settling with partial reflection at the surface. Provision is also made (in Model 4) to account for changes in meteorological structure along the cloud trajectory. A description of the mathematical specifications of the models included in the program is given in Section 3 above.

The multilayer model program is capable of accepting input data from cards from the Preprocessor Program or from input card data sets supplied by the user.

Program output options include:

- Printing of all data inputs
- Printing of the results of all model calculations
- Plotting of maximum centerline concentration, dosage, time-mean concentration and deposition versus distance along the cloud trajectory
- Plotting of concentration, dosage, time-mean concentration and deposition isopleths

A simplified block diagram illustrating major features of the program is s shown in Figure 4-1. A description of the NASA/MSFC Multilayer Diffusion Models Program complete with user's instructions is contained in Appendix B.

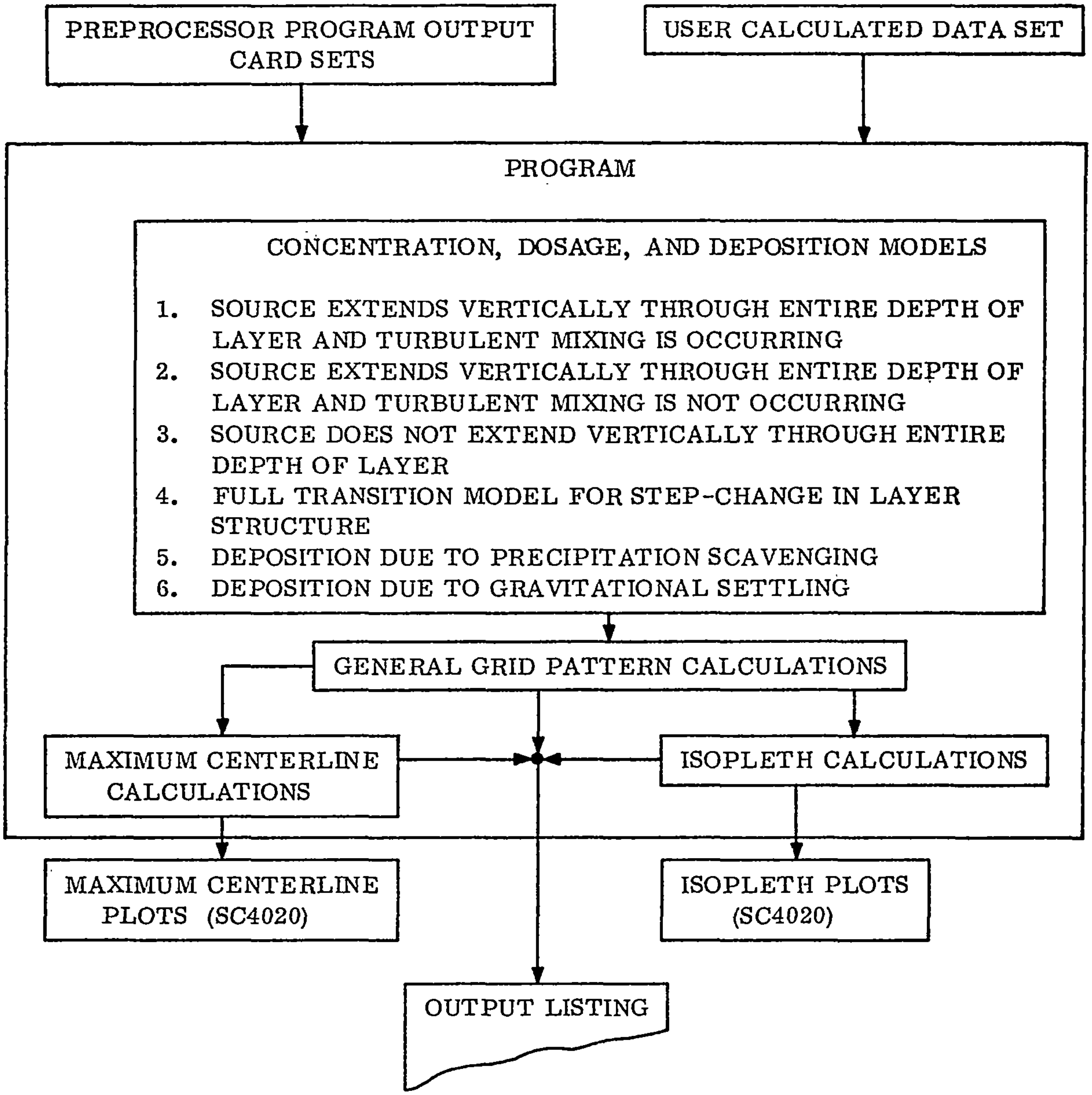


Figure 4-1 Simplified block diagram of the computer program for the NASA/MSFC Multilayer Diffusion Models.

SECTION 5

EXAMPLE CALCULATIONS

Example calculations have been made for both normal and abnormal launches of the Space Shuttle vehicle to illustrate the use of the Preprocessor Program in conjunction with the NASA/MSFC Multilayer Diffusion Models Program - Version 5. The meteorological data used in the example calculations are described in Section 5.1. The results of the calculations for Model 3 are given in Section 5.2 while those for Model 4 are given in Section 5.3. Computer printout of the example calculations is presented in Appendix D.

5.1 METEOROLOGICAL INPUT DATA

Ground-level concentrations and time-mean concentrations of HCl, CO and Al_2O_3 were calculated for normal and abnormal launches of the Space Shuttle using meteorological measurements made at Kennedy Space Center on 21 October 1972. Surface weather maps show that a cold front approached Florida on 19 October and passed the Cape on the morning of 20 October. By 21 October, the cold front was located just south of Florida. Meteorological data from a radiosonde released at 1115 Z on 21 October are given in Table 5-1. Figure 5-1 shows temperature, wind-speed and wind-direction profiles obtained from these data. The temperature profile shows temperature decreasing with height to about 1600 meters above the surface with the more rapid decrease occurring between 300 and 1400 meters. The wind speed increases from 6 meters per second near the surface to 11 meters per second at 750 meters, remains constant between 750 and 1432 meters, and then decreases with height. As shown in Table 5-1, the relative humidity increases with height to 1432 meters and then decreases. From this information, the depth of the surface mixing layer H_m, which is a required input to the Preprocessor Program, was set equal to 1432 meters. The surface density at the time of the radiosonde

TABLE 5-1

RADIOSONDE MEASUREMENTS FOR 1115 Z ON 21 OCTOBER 1972 USED IN THE EXAMPLE CALCULATIONS

Height (meters)	Wind Direction (degrees)	Wind Speed (m sec^{-1})	Temperature (^{o}C)	Pressure (mb)	Relative Humidity (percent)
18	80	6	22.6	1022	57
194	81	8	22.2	1000	57
250	82	9	22.1	994	57
284	82	10	22.0	990	58
500	79	10	19.3	965	65
558	79	10	18.5	959	67
637	78	10	17.8	950	70
750	76	11	16.9	938	74
1000	71	11	14.6	911	86
1098	68	11	13.7	900	93
1135	67	11	13.3	896	94
1250	63	11	12.3	884	97
1432	56	11	10.7	865	97
1500	53	10	10.5	858	90
1577	49	10	10.3	850	79
1716	40	9	9.9	836	55
1750	37	8	10.3	832	55
2000	9	6	12.5	808	49
2259	344	5	11.1	784	44
2500	342	5	9.1	761	54

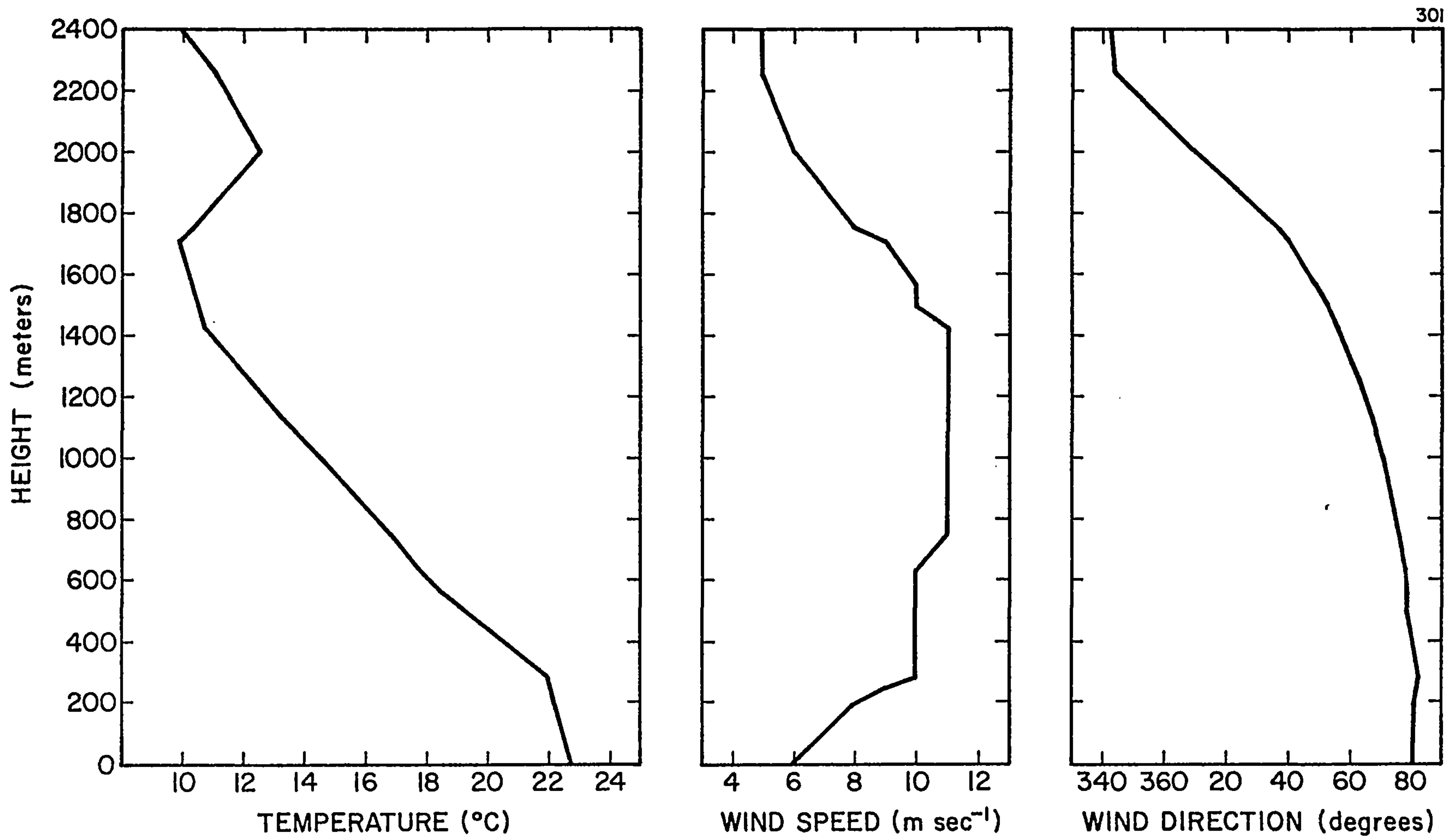


FIGURE 5-1. Vertical profiles of temperature, wind speed and wind direction at Kennedy Space Center for 21 October 1972, 1115 Z.

launch was 1197.07 grams per cubic meter. Bi-directional vane measurements from towers at KSC indicated that the reference standard deviation of the wind azimuth angle σ_{AR} for a ten-minute period was approximately 9 degrees at a height of 18 meters. Other meteorological input parameters required by the Preprocessor Program are supplied by the data given in Table 5-1.

5.2 RESULTS OF CALCULATIONS USING MODEL 3 OF THE NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM

Table 5-2 contains the results of the Preprocessor Program calculations of the Model 3 inputs using Space Shuttle source data and the meteorological inputs for 21 October 1972 at KSC. The cloud stabilization height for a normal launch was calculated using Equation (2-2) while Equation (2-8) was used for the two abnormal launches. Effective heights of the stabilized cloud in the surface mixing layer are based on the relationships given by Equation (2-13). The source dimensions were calculated from Equations (2-11) and (2-12). Figure 5-2 shows the configuration of the exhaust product cloud at the time of stabilization for the normal launch. The stippled area in Figure 5-2 represents the dimensions of the stabilized cloud in the surface mixing layer used in the Model 3 calculations. The position of the stabilized cloud in relation to the launch pad and the time required for cloud stabilization are also given in Table 5-2. The source strengths of the three exhaust products HCl, CO and Al_2O_3 were calculated from Equation (2-32).

The results of the Model 3 calculations of the maximum centerline χ_c and ten-minute average $\chi\{10\ \text{min}\}$ HCl concentrations at the surface for a normal launch, single-engine burn and slow-burn on the pad are shown in Figures 5-3 through 5-5. Figure 5-3 shows, for a normal launch, a calculated peak maximum centerline HCl concentration of 0.8 parts per million (ppm) at a distance of about 12.5 kilometers downwind from the launch pad. The profile of centerline ($y = 0$) ten-minute average HCl concentration, which is obtained by calculating the mean concentration over the ten-minute period of highest concentrations as the cloud passes a point in space, shows a maximum of 0.21 ppm HCl at a distance of 12.5 kilometers from the

TABLE 5-2

MODEL 3 INPUT PARAMETERS PRODUCED BY THE PREPROCESSOR PROGRAM

Model 3 Input Parameters	Type of Launch		
	Normal	Single-Engine Burn	Slow Burn
	a) Cloud Height		
Cloud Stabilization Height, z_m (m)	1790	1910	1772
Effective Height, H (m)	1038	1194	1159
	b) Source Dimensions		
Dimension (m)			
$\sigma_{xo} = \sigma_{yo}$	533	444	412
σ_{zo}	183	111	127
	c) Time of Rise		
Time of Cloud Stabilization t^* (sec)	447	400	452
	d) Cloud Position from Pad at Stabiliation		
Range, R_K (m)	4103	3646	4218
Azimuth, A_K (deg)	235	230	235
	e) Source Strength in the Surface Mixing Layer		
Exhaust Product			
HCl (mg)	8.092×10^9	8.808×10^9	2.555×10^{10}
CO (mg)	1.425×10^{10}	1.551×10^{10}	4.499×10^{10}
Al_2O_3 (mg)	1.802×10^{10}	1.961×10^{10}	5.700×10^{10}

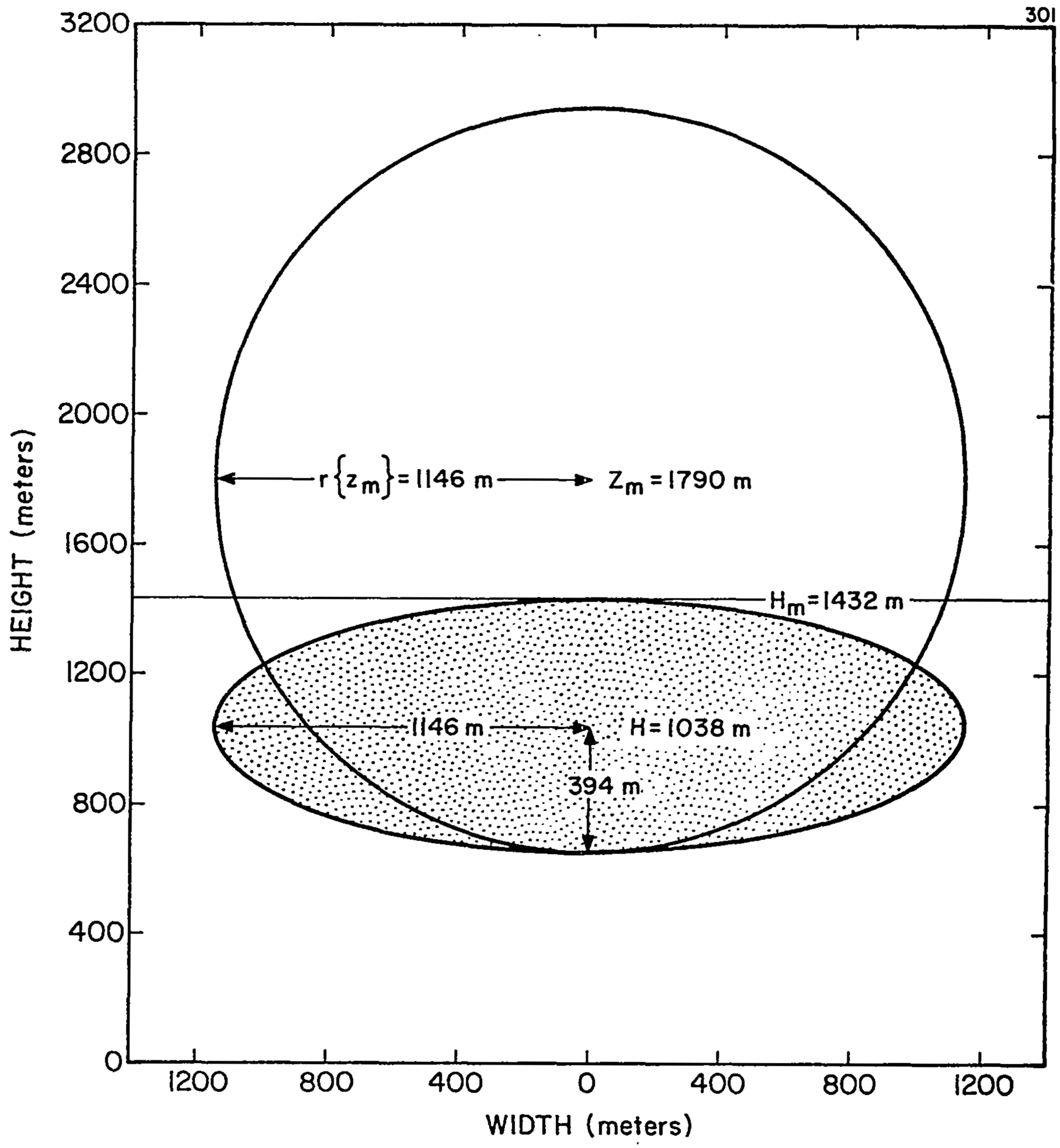


FIGURE 5-2. Configuration of the stabilized cloud of exhaust products used with Model 3 in calculations for a simulated normal launch of a Space Shuttle vehicle on 21 October 1972. Stippled area represents the effective cloud dimensions in the surface mixing layer.

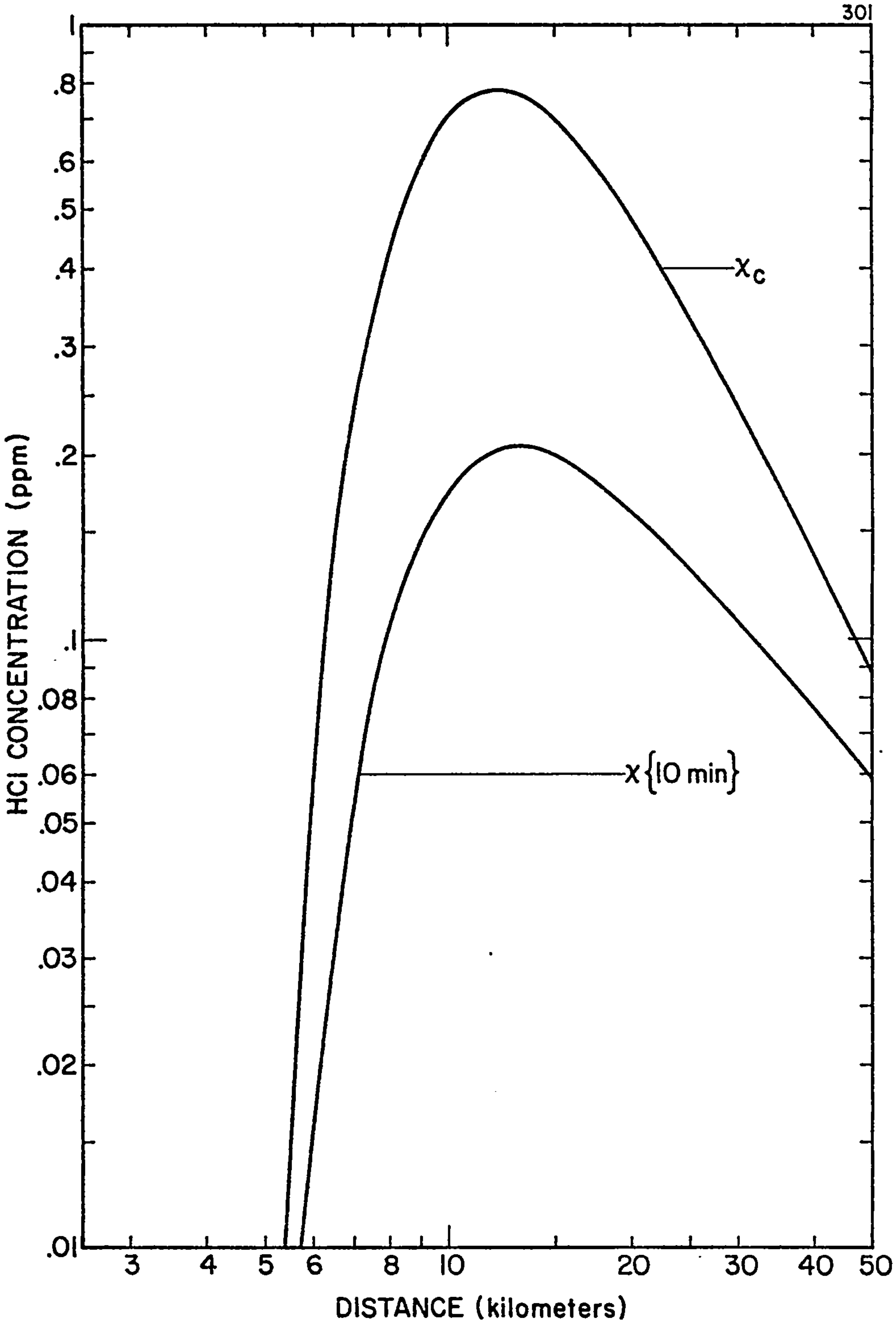


FIGURE 5-3. Maximum centerline χ_c and ten-minute average χ.{10 min} HCl concentrations at ground-level for the simulated normal launch of the Space Shuttle on 21 October 1972 using Model 3.

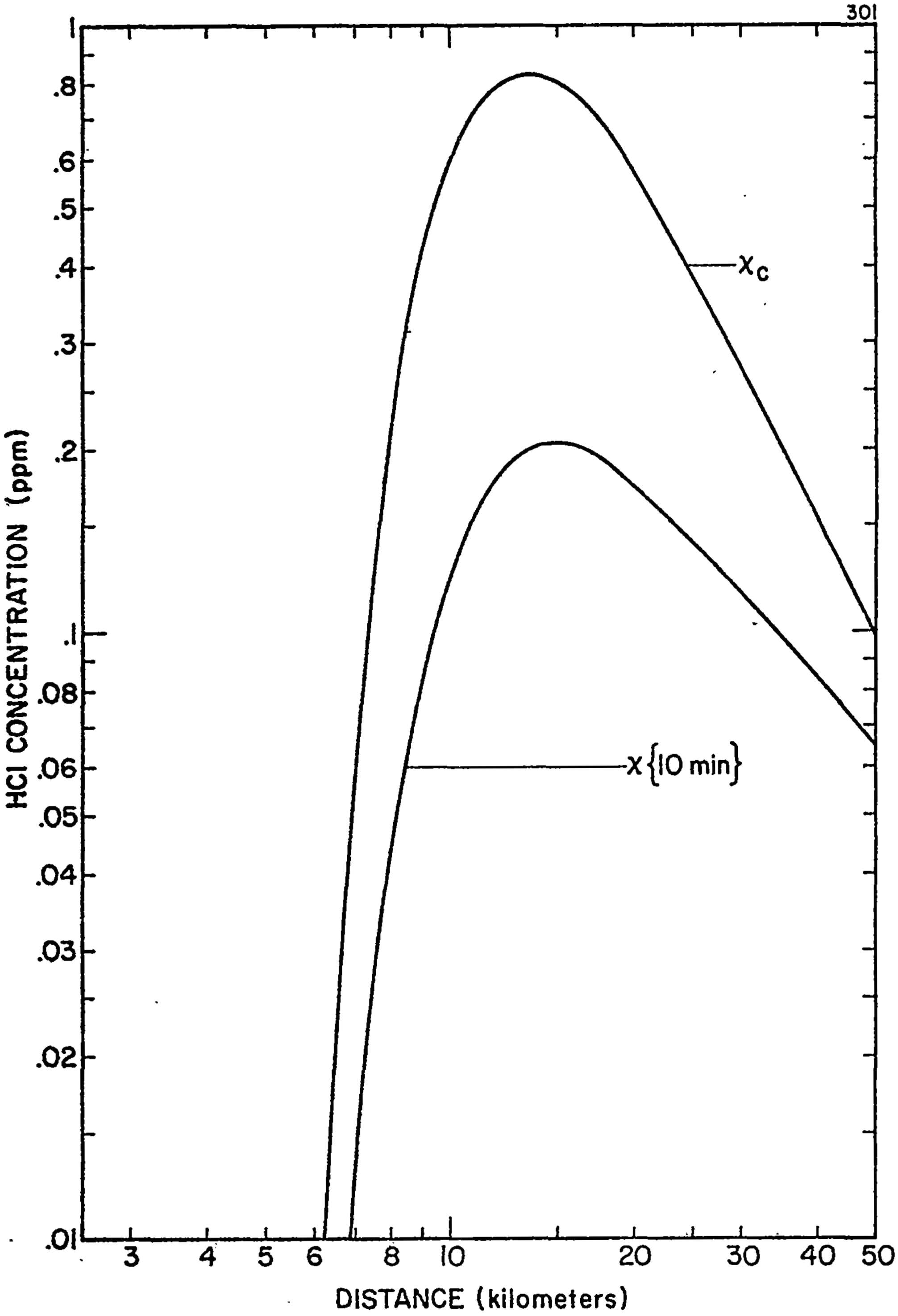


FIGURE 5-4. Maximum centerline χ_c and ten-minute average $\chi\{10\ \text{min}\}$ HCl concentrations at ground-level for the simulated single-engine burn of the Space Shuttle on 21 October 1972 using Model 3.

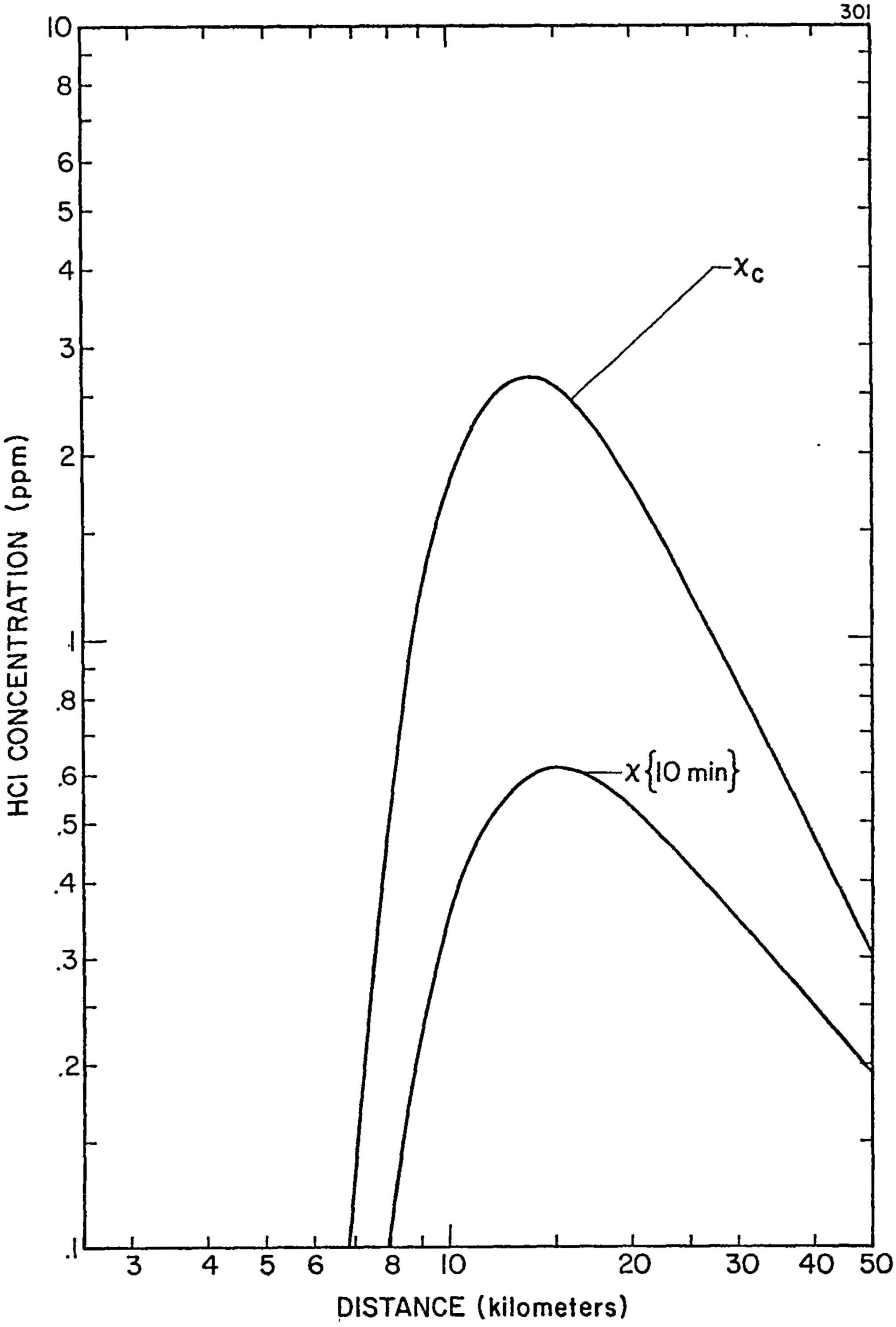


FIGURE 5-5. Maximum centerline χ_c and ten-minute average χ {10 min} HCl concentrations at ground-level for the simulated slow-burn of the Space Shuttle on 21 October 1972 using Model 3.

launch pad. The corresponding peak maximum centerline HCl concentration for a single-engine launch abort, as shown in Figure 5-4, is 0.84 ppm at a distance of about 13.5 kilometers from the launch pad. The maximum ten-minute average concentration of .21 ppm occurs at 15 kilometers from the pad. Comparison of the Model 3 calculations for the slow-burn shown in Figure 5-5 with the corresponding results for a normal launch and single-engine burn shows that ground-level concentrations are greater downwind from the launch pad following the slow-burn abort. The maximum centerline HCl concentration for the slow-burn case is 2.7 ppm HCl at 14 kilometers from the pad and the maximum ten-minute average concentration is 0.62 ppm HCl at 15 kilometers from the pad.

Concentration and dosage profiles and isopleths of concentration and dosage for HCl calculated using Model 3 can be found in the computer listing of example solutions in Appendix D.2.

5.3 RESULTS OF CALCULATIONS USING MODEL 4 OF THE NASA/ MSFC MULTILAYER DIFFUSION MODELS PROGRAM

Table 5-3 presents the results of the Preprocessor Program calculations of inputs to Model 4 calculations of HCl, CO and Al_2O_3 concentrations downwind from normal and abnormal simulated launches of the Space Shuttle vehicle on 21 October 1972 at Kennedy Space Center. The source strength distribution shown in Table 5-3 was calculated following the precedures described in Section 2.3.2 which utilizes Equations (2-27) through (2-34). Source dimensions for Model 4 were calculated using Equation (2-14) and the source positions at the time of cloud stabilization were calculated from Equations (2-15) through (2-26). Figure 5-6 shows the configuration of the exhaust cloud at the time of stabilization for the normal launch of the Space Shuttle used in the Model 4 calculations. The abscissa of Figure 5-6 is the range from the launch pad without consideration of the offset due to differences in the azimuth bearings from the launch pad. The calculated cloud stabilization heights and stabilization times for Model 4 are identical to those shown in Table 5-2 for Model 3 calculations.

TABLE 5-3

PREPROCESSOR PROGRAM CALCULATED SOURCE STRENGTHS, SOURCE DIMENSIONS AND SOURCE POSITION FOR MODEL 4

Layer No.	Height of Layer Top (m)	Source Strength (mg)			Source Dimensions (m)	Source Position	
		HCl	CO	Al_2O_3	$\sigma_{xo} = \sigma_{yo}$	Range R_K (m)	Azimuth A_K (m)
a) Normal Launch							
1	194	4.423×10^7	7.787×10^7	9.849×10^7	533	40	261
2	250	1.788×10^7	3.147×10^7	3.981×10^7	533	67	261
3	284	1.383×10^7	2.436×10^7	3.080×10^7	533	88	261
4	500	1.737×10^8	3.059×10^8	3.869×10^8	533	279	261
5	558	8.535×10^7	1.503×10^8	1.901×10^8	533	343	260
6	637	1.565×10^8	2.756×10^8	3.486×10^8	533	439	260
7	750	3.305×10^8	5.818×10^8	7.359×10^8	533	601	259
8	1000	1.407×10^9	2.477×10^9	3.133×10^9	533	1054	257
9	1098	9.012×10^8	1.587×10^9	2.007×10^9	533	1260	256
10	1135	4.021×10^8	7.079×10^8	8.954×10^8	533	1341	255
11	1250	1.482×10^9	2.609×10^9	3.300×10^9	533	1610	253
12	1432	3.078×10^9	5.419×10^9	6.854×10^9	533	2084	250

TABLE 5-3

PREPROCESSOR PROGRAM CALCULATED SOURCE STRENGTHS, SOURCE DIMENSIONS AND SOURCE POSITION FOR MODEL 4

(Continued)

Layer No.	Height of Layer Top (m)	Source Strength (mg)			Source Dimensions (m)	Source Position	
		HCl	CO	Al_2O_3	$\sigma_{xo} = \sigma_{yo}$	Range R_K (m)	Azimuth A_K (m)
b) Single-Engine Burm							
1	194	3.500×10^6	6.162×10^6	7.794×10^6	444	49	261
2	250	2.316×10^6	4.078×10^6	5.158×10^6	444	75	261
3	284	2.047×10^6	3.604×10^6	4.558×10^6	444	93	261
4	500	3.907×10^7	6.878×10^7	8.700×10^7	444	262	261
5	558	2.608×10^7	4.592×10^7	5.808×10^7	444	315	260
6	637	5.692×10^7	1.002×10^8	1.268×10^8	444	394	260
7	750	1.517×10^8	2.672×10^8	3.379×10^8	444	522	259
8	1000	9.836×10^8	1.732×10^9	2.190×10^9	444	867	257
9	1098	8.452×10^8	1.488×10^9	1.882×10^9	444	1019	256
10	1135	4.214×10^8	7.419×10^8	9.384×10^8	444	1078	255
11	1250	1.759×10^9	3.097×10^9	3.917×10^9	444	1272	254
12	1432	4.517×10^9	7.953×10^9	1.006×10^{10}	444	1605	251

TABLE 5-3

PREPROCESSOR PROGRAM CALCULATED SOURCE STRENGTHS, SOURCE DIMENSIONS AND SOURCE POSITION FOR MODEL 4

(Continued)

Layer No.	Height of Layer Top (m)	Source Strength (mg)			Source Dimensions (m)	Source Position	
		HCl	CO	Al_2O_3	$\sigma_{xo} = \sigma_{yo}$	Range R_K (m)	Azimuth A_K (m)
c) Slow Burn							
1	194	8.035×10^6	1.415×10^7	1.789×10^7	412	61	261
2	250	5.793×10^6	1.020×10^7	1.290×10^7	412	93	261
3	284	5.234×10^6	9.215×10^6	1.167×10^7	412	117	261
4	500	1.073×10^8	1.889×10^8	2.390×10^8	412	328	261
5	558	7.463×10^7	1.314×10^8	1.662×10^8	412	395	260
6	637	1.662×10^8	2.927×10^8	3.702×10^8	412	494	260
7	750	4.527×10^8	7.971×10^8	1.008×10^9	412	655	259
8	1000	2.988×10^9	5.261×10^9	6.654×10^9	412	1092	257
9	1098	2.553×10^9	4.496×10^9	5.686×10^9	412	1286	256
10	1135	1.262×10^9	2.223×10^9	2.811×10^9	412	1361	255
11	1250	5.187×10^9	9.132×10^9	1.155×10^{10}	412	1611	254
12	1432	1.274×10^{10}	2.243×10^{10}	2.837×10^{10}	412	2044	251

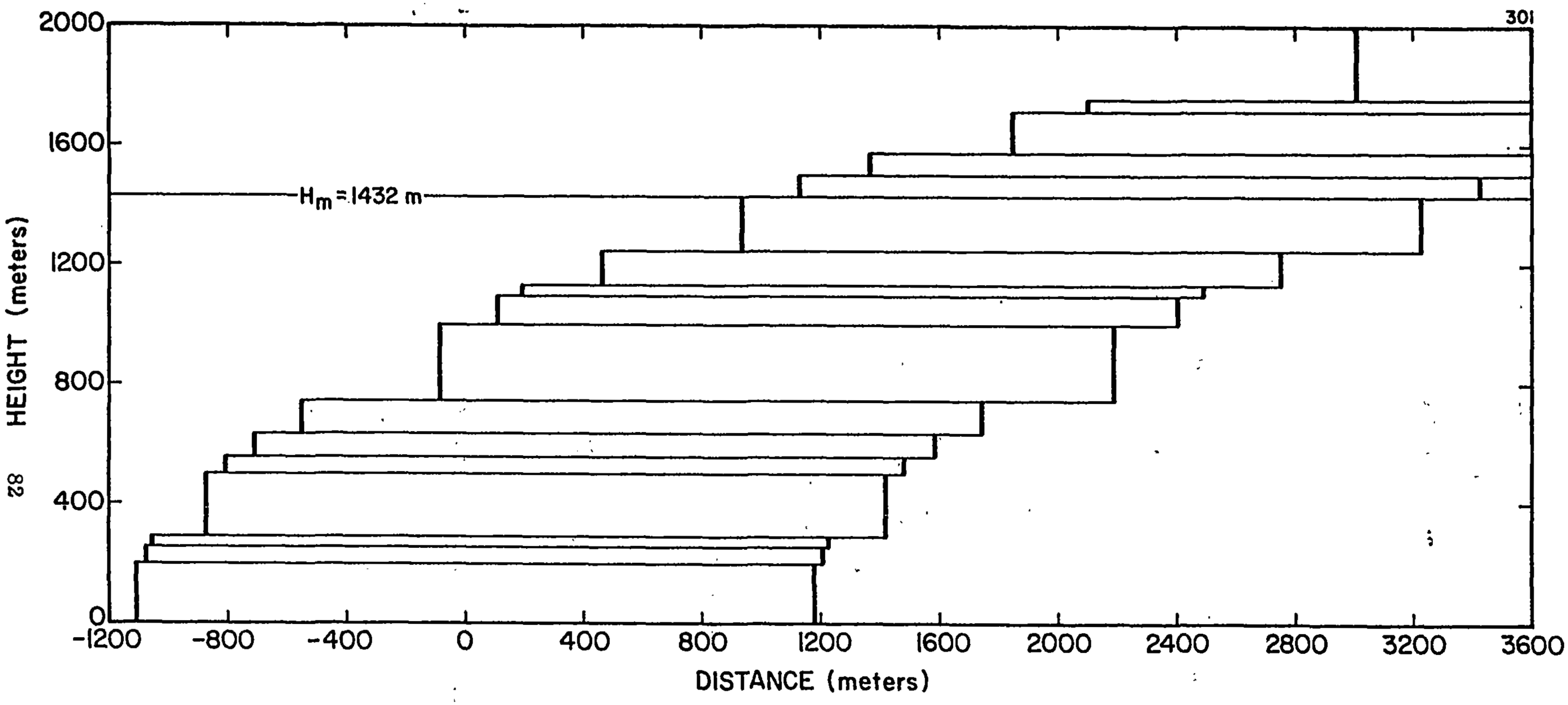


FIGURE 5-6. Configuration of the stabilized cloud of exhaust products used with Model 4 in calculations for a simulated normal launch of a Space Shuttle vehicle at Kennedy Space Center on 21 October 1972.

The results of the Model 4 calculations of the maximum centerline and ten-minute average HCl concentrations at the surface downwind from a simulated normal launch, single-engine burn and slow burn of the Space Shuttle vehicle are respectively shown in Figures 5-7, 5-8 and 5-9. The Model 4 HCl peak centerline concentrations χ_c are 0.6 ppm, .74 ppm and 2.25 ppm for a normal launch, single-engine burn and slow burn on the pad. All of these occur at a distance of approximately 12.5 kilometers downwind from the launch pad. Similarly, the maximum ten-minute average HCl concentrations are .17 ppm, 0.18 and 0.53 ppm respectively for a normal launch, single-engine burn and slow burn on the pad. All of these occur at a distance of approximately 15 kilometers downwind from the launch pad.

Table 5-4 shows a comparison of the results of the Model 4 calculations with those obtained from Model 3. The Model 3 maximum concentrations are 15 to 30 percent greater than the maximum concentrations calculated using Model 4. On the other hand, Model 4 predicts higher concentrations than Model 3 at distances close to the launch pad. These differences are a consequence of the two source configurations as shown in Figure 5-2 (Model 3) and Figure 5-6 (Model 4). Until accurate measurements of the vertical distribution of exhaust products in the stabilized cloud become available, model calculations of concentrations and dosages close to the launch pad are subject to this type of uncertainty. At longer downwind distances from the launch pad, as comparison of the Model 3 and Model 4 calculations shows, there is close agreement in the predicted concentrations because the effects of differing assumptions regarding the source configuration are small.

Calculated concentration and dosage profiles and isopleths of concentration and dosage for HCl obtained by using Model 4 are also presented in the computer listing of example solutions in Appendix D.2.

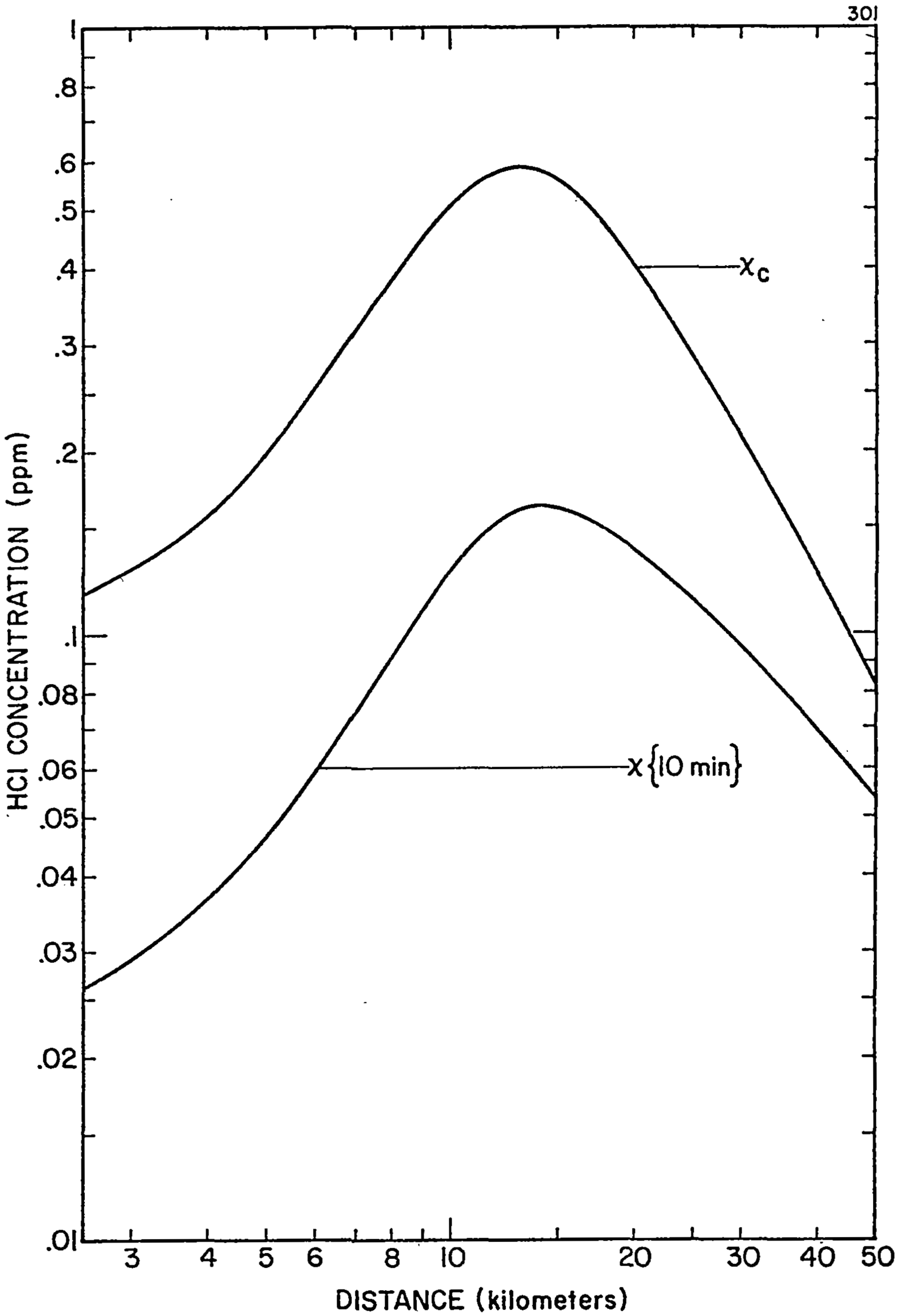


FIGURE 5-7. Maximum centerline χ_c and ten-minute average $\chi\{10\ \text{min}\}$ HCl concentrations at ground-level for the simulated normal launch of the Space Shuttle on 21 October 1972 using Model 4.

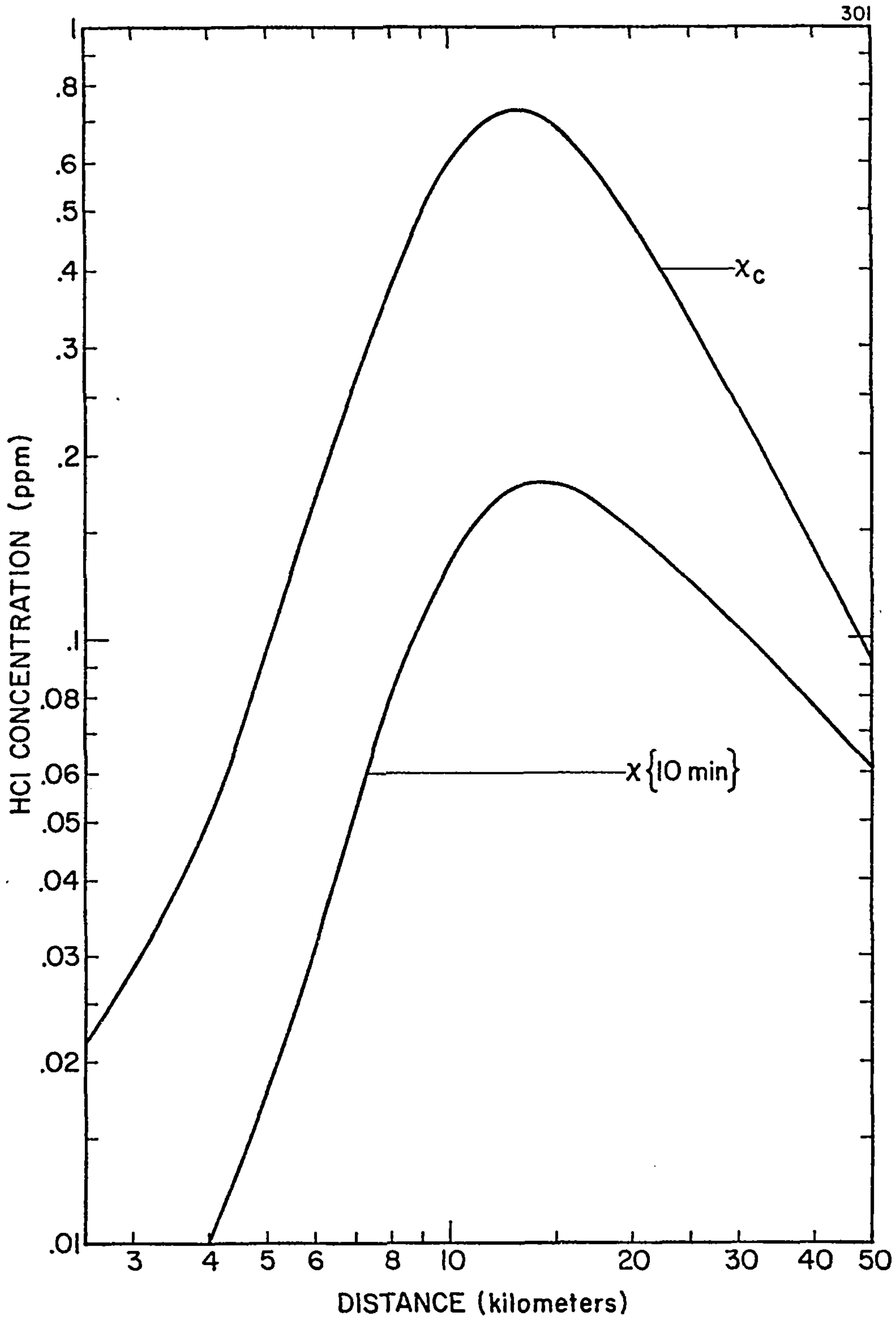


FIGURE 5-8. Maximum centerline χ_c and ten-minute average $\chi\{10\ \text{min}\}$ HCl concentrations at ground-level for the simulated single-engine burn of the Space Shuttle on 21 October 1972 using Model 4.

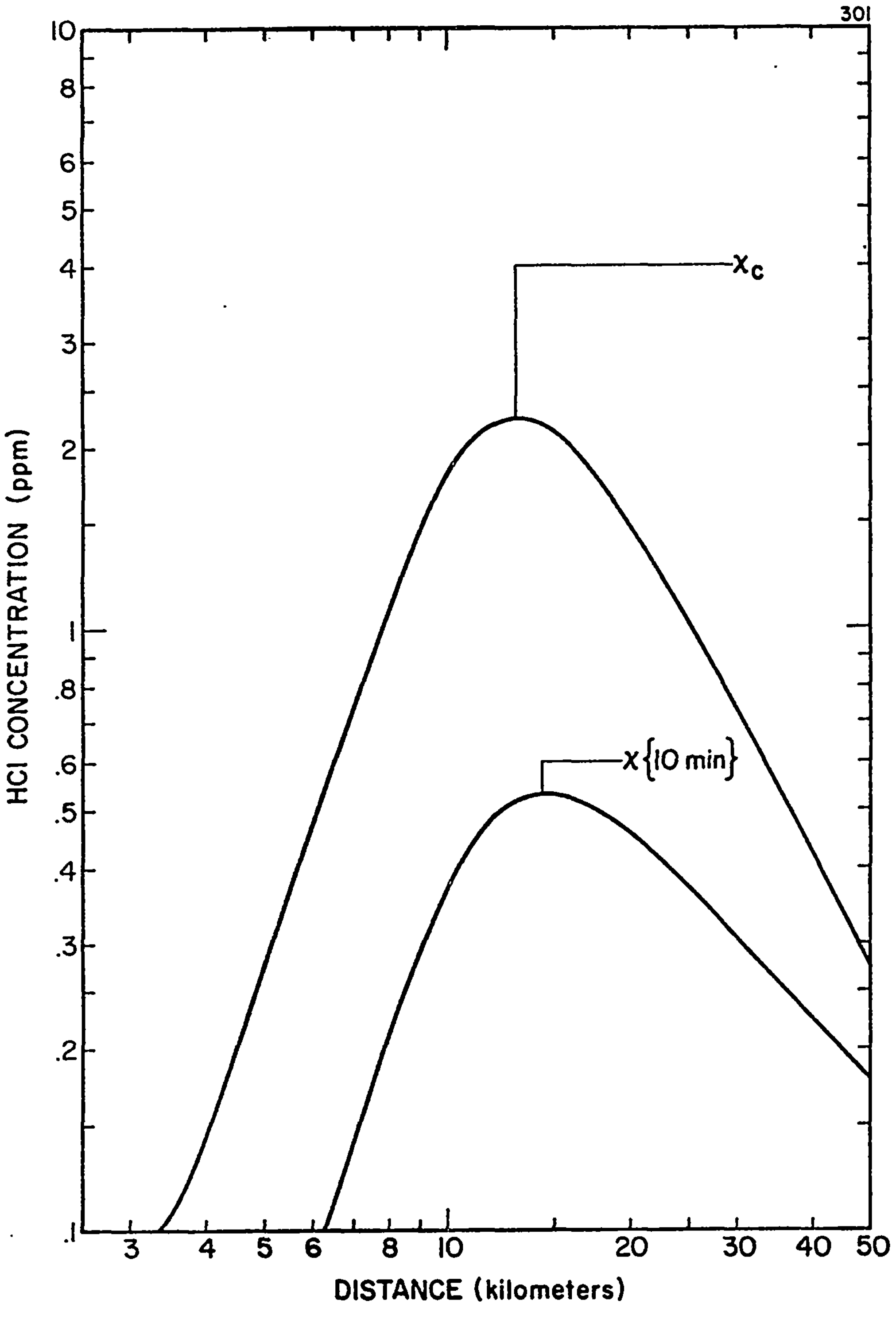


FIGURE 5-9. Maximum centerline χ_c and ten-minute average χ {10 min} HCl concentrations at ground-level for the simulated slow-burn of the Space Shuttle on 21 October 1972 using Model 4.

TABLE 5-4

COMPARISON OF PEAK MAXIMUM CENTERLINE χ_c AND TEN-MINUTE AVERAGE HCl CONCENTRATIONS (ppm) FROM MODEL 3 AND MODEL 4 CALCULATIONS

Model Number	Type of Launch					
	Normal		Single-Engine Burn		Slow Burn	
	χ_c	$\chi\{10\text{ min}\}$	χ_c	$\chi\{10\text{ min}\}$	χ_c	$\chi\{10\text{ min}\}$
3	0.80	0.21	0.84	0.21	2.7	0.62
4	0.60	0.17	0.74	0.18	2.3	0.53

REFERENCES

Abramowitz, M. and J. A. Stegun (ed), 1964: Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables. NBS Applied Mathematics, Series 55, U. S. Government Printing Office, 1046.

Briggs, G. A., 1969: Plume Rise. Available as TID-25075 from Clearinghouse for Federal Scientific and Technical Information, Springfield, Va.

Briggs, G. A., 1970: Some recent analysis of plume rise observations. Paper ME-8E presented at the Second International Clean Air Congress, Washington, D. C., Dec. 6-11, 1970.

Cicerone, R. J., D. H. Stedman and R. S. Stolarski, 1973: Contract for study of potential environmental proglems due to space shuttle operations. Progress Report for August-September 1973 under Contract NAS8-28294, NASA, George C. Marshall Space Flight Center, Huntsville, Alabama 35812.

Cramer, H. E., et al., 1964: Meteorological prediction techniques and data system. Final Rpt. under Contract DA-42-007-CML-552, GCA Tech. Rpt. 64-3-G, U. S. Army Dugway Proving Ground, Dugway, Utah, 252.

Cramer, H. E., et al., 1967: Determination of hazard prediction procedures. GCA Tech. Rpt. 67-8-G, Final Rpt. under Contract No. DA-42-007-AMC-359(R), U. S. Army Dugway Proving Ground, Dugway, Utah.

Dumbauld, R. K., et al., 1970: Handbook for estimating toxic fuel hazards. Final Rpt. under Contract NAS8-21453, NASA Rpt. CR-61326.

Dubmauld, R. K., J. R. Bjorklund and J. F. Bowers, 1973: NASA/MSFC multilayer diffusion models and computer program for operational prediction of toxic fuel hazards. NASA Contract Rpt. NASA CR-129006, NASA - George C. Marshall Space Flight Center, Alabama.

Englemann, R. J. and W. G. N. Slinn, 1970: Precipitation Scavenging (1970). Proceedings of a Symposium held at Richland, Washington, June 2-4, 1970.

Greville, T. N. E., 1969: Theory and Applications of Spline Functions, Academic Press, New York, N. Y.

Knutson, et al., 1974: Washout coefficients for scavenging of rocket exhaust HCl by rain. Paper presented at the Sixth Conference on Aerospace and Aeronautical Meteorology, El Paso, Texas, Nov. 12-15, 1974.

Pasquill, F., 1974: Atmospheric Diffusion, Second Edition, John Wiley and Sons, New York, N. Y.

Pellett, G. L., 1974: Washout of HCl and application to solid rocket exhaust clouds. Paper presented at Precipitation Scavenging Symposium (1974), Champaign, Illinois, Oct. 14-18, 1974.

Tabata, S., 1973: A simple but accurate formula for the saturation vapor pressure over liquid water. JAM, 12, p. 1410.

Tyldesley, J. B. and C. E. Wallington, 1965: The effect of wind shear and vertical diffusion on horizontal dispersion. Quart. J. Roy. Met. Soc., 91, 158-174.

APPENDIX A

USER INSTRUCTIONS FOR THE PREPROCESSOR PROGRAM
FOR USE WITH THE NASA/MSFC MULTILAYER DIFFUSION MODELS
COMPUTER PROGRAM -VERSION 5

The Preprocessor Program produces a complete set of data decks for input to the NASA/MSFC Multilayer Diffusion Models Computer Program - Version 5. The Program is specifically designed for use with launches of Space Shuttle, Titan III C, Delta-Thor, and Minuteman II vehicles. The data decks produced on option by this Program include a complete card deck for each of the four pollutants HCl, CO, CO_2 and Al_2O_3 for Models 3 and/or 4 in the NASA/MSFC Multilayer Diffusion Models Computer Program - Version 5.

The Preprocessor Program is written in FORTRAN IV and uses approximately 9700_{10} locations of computer core storage. The Program requires card input, print and punch output and executes in less than 2 seconds on a UNIVAC 1108 computer. A complete computer listing is given in Appendix C.1 below.

A.1 PROGRAM INPUT PARAMETERS

The Preprocessor Program requires the input of the following meteorological parameters:

σ_{AR}	-	Standard deviation of the wind azimuth angle in degrees measured at the first reference height z_1 over a 10 minute time period.
ρ	-	Ambient air density in grams per cubic meter measured at z_1.
z	-	Height in feet or meters at which the meteorological measurements are taken.

θ	-	Wind direction in degrees at z.
u	-	Wind speed in knots or meters per second at z.
T	-	Ambient air temperature in degrees Celsius at z.
P	-	Ambient air pressure in millibars at z.
RH	-	Relative humidity in percent at z.

The Preprocessor Program also requires control information indicating: (1) vehicle type, (2) whether the computer run is for a normal or abnormal launch, (3) whether z is in feet or meters, (4) whether u is in knots or meters per second, (5) height of the surface mixing layer which must coincide with one of the z inputs above, and (6) the model being used and the pollutants for which data decks will be produced.

A.2 PROGRAM INPUT DATA CARD SEQUENCE

The first card in the input data deck contains general case titling information and is used for a page heading in the Preprocessor print output and is also punched in the output data deck for input to the Multilayer Models Program. The second input card contains control information and σ_{AR} and ρ at the surface. An example set of data input cards is shown in Figure A-1. The example problem is described in Section 5 in the main body of this report.

Data Card 1:

Columns 1 - 72	-	General data set titling information. If input as blanks the program will use the last information input

Data Card 2:

Columns 2-4	-	Punch these characters indicating the vehicle type. If left blank Titan III C is assumed. TTN is the Titan III C vehicle; STL is the Space Shuttle vehicle; DTH is the Delta-Thor vehicle; MIN is the Minuteman II vehicle.
Columns 5-7	-	Punch YES or leave blank if the run is for a normal launch. Punch NØ1 if the run is for an abnormal launch where a single engine burns on the launch pad. Not produced for the Delta-Thor and Minuteman II vehicles. Punch NØ2 if the run is for an abnormal launch where a slow burn on the pad occurs. (Ø is alphabetic)
Column 8	-	Punch M or leave blank if the heights z are in meters. Punch F if the heights are in feet.
Column 9	-	Punch M or leave blank if the wind speed u is in meters per second. Punch K if the wind speed is in knots.
Columns 10-45	-	Punch the date of the meteorological case or any case identification information (optional).
Columns 46-55	-	Punch σ_{AR} (see Figure A-1).
Columns 56-65	-	Punch ρ (see Figure A-1).
Column 67	-	Punch a 1 if output for Model 4 is desired; leave blank if not.
Column 68	-	Punch a 1 output for Model 3 is desired; leave blank if not.

Data Card 2 (Continued):

Column 69	-	Punch a 1 if output for HCl is desired; leave blank if not.
Column 70	-	Punch a 1 if output for CO is desired; leave blank if not.
Column 71	-	Punch a 1 if output for Al_2O_3 is desired; leave blank if not.
Column 72	-	Punch a 1 if output for CO_2 is desired; leave blank if not. Produced only for the Titan III C vehicle.
Column 73	-	Punch a 1 if cloud the trajectory range and azimuth bearing are to be calculated; leave blank if not.

Data Cards 3 to N-1:

Columns 1-10	-	Punch z (see Figure A-1).
Columns 11-20	-	Punch θ (see Figure A-1).
Columns 21-30	-	Punch u (see Figure A-1).
Columns 31-40	-	Punch T (see Figure A-1).
Columns 41-50	-	Punch P (see Figure A-1).
Columns 51-60	-	Punch RH (see Figure A-1).
Column 80	-	Punch and asterisk (*) if this height z coincides with the surface mixing layer height; otherwise, leave blank. If not punched on any card, the program assumes the last z input to be the mixing layer height (See Figure A-1).

Data Card N:

The last card in the input data deck must be a blank card. Multiple case capability is provided by punching a non-blank character in column 80 of this card. If column 80 is non-blank the Program expects another complete case, otherwise the program stops.

The Preprocessor Program assumes the decimal point in all of the above meteorological parameters to be between the sixth and seventh columns of the 10-column field if it is not punched. For example, if the 10-column field 41 to 50 contained the number ΔΔ121033ΔΔ (Δ is a blank), the program would interpret it as 1210.32. To avoid improper alignment of a data value, the decimal should be punched. Also, unless it is certain that the cloud rise height will not exceed the mixing layer height, it is recommended that at least three heights z be input above the height coinciding with the mixing layer height. If the cloud rise height exceeds the last height input, the program will stop and ask for data at greater heights.

A.3 COMPUTER PROGRAM PUNCH OUTPUT

The Preprocessor Program will punch a complete card deck for direct input to the NASA/MSFC Multilayer Models - Version 5. The first card of each of the ouput data decks contains $NAM2 and the last card contains $END. The possible data decks punched are:

Inputs for:	(1)	HCl	-	Model 4
	(2)	CO	-	Model 4
	(3)	CO_2	-	Model 4
	(4)	Al_2O_3	-	Model 4
	(5)	HCl	-	Model 3
	(6)	CO	-	Model 3
	(7)	CO_2	-	Model 3
	(8)	Al_2O_3	-	Model 3

EXAMPLE SPACE SHUTTLE NORMAL LAUNCH							
STLYESMMK	SC 21 OCT	72.		9	. 119	7.07 1111	111
18.	80.	6.	22.6	1022.	57.		
194.	81.	8.	22.2	1000.	57.		
250.	82.	9.	22.1	993.66	57.		
284.	82.	10.	22.0	989.79	58.		
500.	79.	10.	19.3	965.4	65.		
558.	79.	10.	18.5	958.94	67.		
637.	78.	10.	17.8	950.	70.		
750.	76.	11.	16.9	937.71	74.		
1000.	71.	11.	14.6	910.61	86.		
1098.	68.	11.	13.7	900.	93.		
1135.	67.	11.	13.3	896.25	94.		
1250.	63.	11.	12.3	884.09	97.		
1432.	56.	11.	10.7	865.13	97.		*
1500.	53.	10.	10.5	858.17	90.		
1577.	49.	10.	10.3	850.	79.		
1716.	40.	9.	9.9	836.31	55.		
1750.	37.	8.	10.3	832.87	55.		
2000.	9.	6.	12.5	808.34	49.		
2259.	344.	5.	11.1	783.69	44.		
2500.	342.	5.	9.1	761.39	54.		
BLANK CARD							

FIGURE A-1. Example computer program input for a normal launch of the Space Shuttle vehicle. The heights z are in meters and the wind speeds are in meters per second. The top of the mixing layer is at 1432 meters and the last card is a blank indicating end of data.

A.4 COMPUTER OUTPUT FOR THE EXAMPLE DATA

The Preprocessor Program first prints constant model parameters for the vehicle and prints the input data deck for verification. A complete deck for each selected pollutant HCl, CO, CO_2 and Al_2O_3 is printed and punched for Models 3 and/or 4 of the Multilayer Model. Also, the program lists all calculated parameters for Models 3 and/or 4 in a tabularized form for verification of Program calculations. A complete output listing for the example case shown in Figure A-1 is given in Appendix D.1.

A.5 LINKAGE DIAGRAM OF THE PREPROCESSOR PROGRAM

Figure A-2 shows the subroutine linkage diagram for the Preprocessor Program. Each line terminating at a subroutine name represents a subroutine call. The Program also references the FORTRAN library functions ACØS, CØS, SIN, ATAN2, EXP not shown in Figure A-2. A description of the subroutines shown in Figure A-2 is given in Section A.6.

A.6 DESCRIPTION OF PROGRAM SUBROUTINES

Subroutine VHICLE is the main controlling program in the Preprocessor. This routine sets the constants for the particular vehicle desired, inputs all data, converts inputs into proper units and controls all calculations and output.

Subroutine DIM34 calculates the source dimensions for Models 3 and/or 4 and calculates the effective cloud height (see Section 2, Equations (2-11) through (2-14)).

Subroutine PLUME1 calculates the plume rise for instantaneous sources (see Section 2.2.1).

Subroutine PLUME2 calculates the plume rise for continuous sources (see Section 2.2.2).

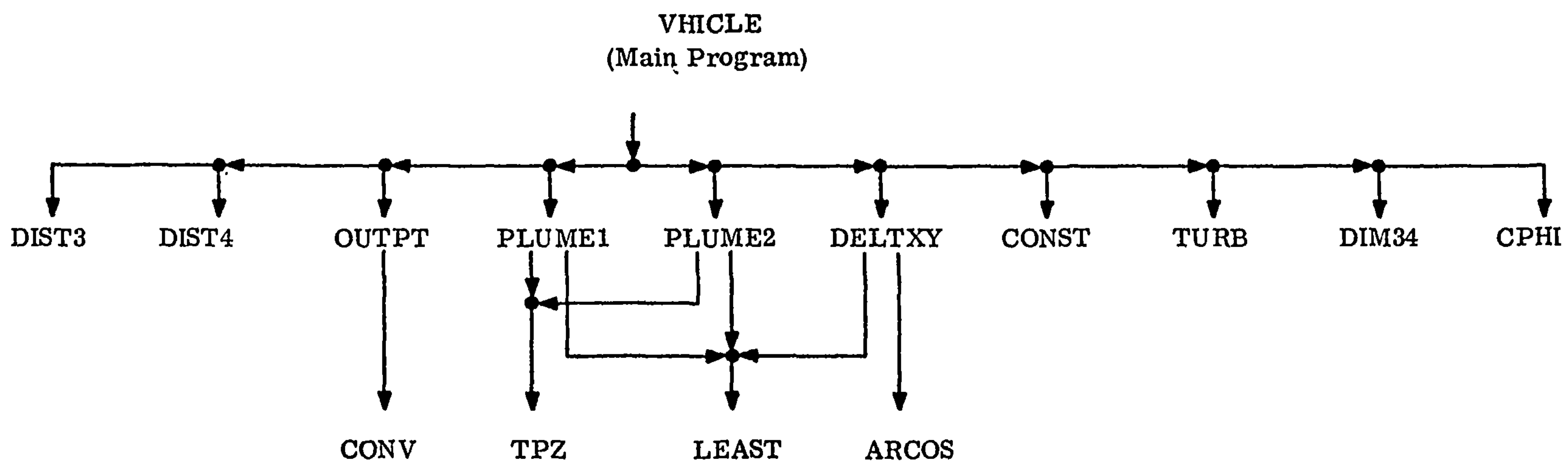


FIGURE A-2. Linkage diagram of the subroutines in the Preprocessor Program.

Subroutine DELTXY calculates the cloud trajectory (see Section 2, Equations (2-15) through (2-26)).

Subroutine LEAST calculates the lapse rate of virtual potential temperature using a least-squares fit (Equation (2-7)).

Subroutine CONV converts the output source strength into a form suitable for punching variable real numbers in a fixed format.

Subroutine TURB inserts the turbulence parameters σ_A and σ_E in the output data and calculates $t_K = t^*$ (Equation (2-4)).

Subroutine TPZ is a linear interpolation function.

Subroutine CPHI is used in calculating the virtual potential temperature (Equation (2-7)).

Subroutine DIST4 calculates the source distribution for Model 4 (see Section 2.3.2).

Subroutine DIST3 calculates the source distribution for Model 3 (see Section 2.3.2).

Subroutine OUTPT prints and punches the output data in a namelist format suitable for input to the NASA/MSFC Multilayer Diffusion Models - Version 5.

Subroutine CONST prints the input parameters and resultant calculations in a format suitable for error check and calculation review.

Subroutine ARCOS is a function reference to ACOS, the UNIVAC 1108 arc-cosine function. It is referenced in this way because the IBM 7044 at MSFC uses ARCOS for the arc-cosine function and program conversion from the UNIVAC 1108 to the IBM 7044 is easily completed by removal of this entire subroutine.

APPENDIX B

USER INSTRUCTIONS FOR THE NASA/MSFC MULTILAYER DIFFUSION MODELS COMPUTER PROGRAM - VERSION 5

B.1 PROGRAM DESCRIPTION

The NASA/MSFC Multilayer Diffusion Models Program - Version 5 is designed to calculate patterns of:

- Concentration
- Dosage
- Time-mean concentration
- Average cloud concentration
- Time of cloud passage
- Ground-level deposition due to precipitation scavenging
- Ground-level deposition due to gravitational settling

Program options include the calculation of concentration, dosage and time-mean concentration with partial reflection, with time dependent decay, and/or with depletion due to precipitation scavenging. Also, the Program is capable of calculating ground-level gravitational deposition with partial reflection of material at the surface. Other Program options include the printing of all data inputs, the printing of all model calculations, the plotting of concentration, dosage and/or time-mean concentration on the printer page, and the plotting of concentration, dosage and/or time-mean concentration on the SC4020 plotter at Marshall Space Flight Center.

The NASA/MSFC Multilayer Diffusion Models Program - Version 5 is written in FORTRAN IV and is designed for use on the UNIVAC 1108 computer at Marshall Space Flight Center, Huntsville, Alabama. The Program requires 33000_{10} locations of core storage on the UNIVAC 1108 computer. The FORTRAN source listing is shown in Appendix C. 2 and a complete list of Program input parameters and Program options is given in Section B. 2 and B. 3 below. Also, a linkage diagram of the Program subroutines is shown in Section B. 6 and a description of the Program subroutines is given in Section B. 7.

B. 2 PROGRAM INPUT PARAMETERS

This section gives a complete description of all Program input parameters. A condensed table of the input parameters is given in Section B. 3. The data input format is given in Section B. 4 and an example input coding sheet is given in Section B. 5.

NAMCAS - 72 Hollerith characters of general case identification information. This information is printed in addition to the adjusted cloud stabilization height, range and azimuth bearing and the date and time of the run as a title page to the output listing.

TESTNO - The first 36 Hollerith characters (TESTNO (1) - TESTNO (6)) contain the meteorological case information. This information is printed in the page heading and plot titles following the words "THE METEOROLOGICAL CASE IS".

Characters 37 through 60 (TESTNO (7) - TESTNO (10)) contain the name of the rocket vehicle for use in the page heading. (e. g. TITAN IIIC)

Characters 61 through 72 (TESTNO (11) - TESTNO (12)) contain the name of the pollutant only if it is not HCl, CO, CO_2, or Al_2O_3. (e. g. NO_x)

NPS - This parameter is used to indicate multiple cases.

If NPS is set to 0, the Program assumes there is another case to follow.

If NPS is set to 1, the Program assumes this is the last case to process.

ISKIP - Program control option array.

ISKIP (1) - This option, if set non-zero, indicates patterns of concentration, dosage, time-mean concentration, deposition, etc. are to be calculated and printed on the polar reference grid system defined by XX and YY below. The grid system origin is the vehicle launch site and all calculation distances are relative to the origin.

ISKIP (2) - This option, if set non-zero, is used to calculate maximum centerline values of concentration, dosage, time-mean concentration, and/or deposition along the cloud trajectory relative to the launch site.

If ISKIP (2) is set equal to 1, the model calculations are printed.

If ISKIP (2) is set equal to 2, the model calculations are plotted.

If ISKIP (2) is set equal to 3, the model calculations are both printed and plotted.

The maximum centerline concentration, dosage, time-mean concentration and deposition are determined by the use of a spline function. At each radial distance (XX) from the origin, the Program determines a curve via the cubic spline that passes through each angular (azimuth bearing YY) grid coordinate with the calculated maximum roughly in the midpoint of the curve. The Program will then determine the maximum value and output, the range and azimuth bearing to that maximum (see Section B. 7, Subroutine Spline).

ISKIP (3) - This option, if set non-zero, is used to calculate isopleths of concentration, dosage, time-mean concentration and/or deposition.

If ISKIP (3) is set equal to 1, the isopleths are printed.

If ISKIP (3) is set equal to 2, the isopleths are plotted.

If ISKIP (3) is set equal to 3, the isopleths are both printed and plotted.

ISKIP (4) - This option is used only with the calculation of ground-level precipitation deposition (Model 5).

If ISKIP (4) is set non-zero, the maximum possible ground-level precipitation deposition is calculated at points downwind from the cloud position at the time of the start of precipitation (TIM1). These calculations are independent of the elapsed time from TIM1 to the calculation point.

If ISKIP (4) is set equal to zero, the calculated precipitation deposition at points downwind from the cloud position at time TIM1 is dependent upon the elapsed time from TIM1 to the points.

ISKIP (5) - This option controls the pollutant name and units printed in the page heading and plot legend:

If ISKIP (5) is set equal to 1, the units of calculated HCl concentration are in parts per million (ppm) and dosage units are in parts per million seconds.

If ISKIP (5) is set equal to 2, the units of calculated CO concentrations are in parts per million (ppm) and dosage units are in parts per million seconds.

If ISKIP (5) is set equal to 3, the units of calculated CO_2 concentration are in parts per million (ppm) and dosage units are in parts per million seconds.

If ISKIP (5) is set equal to 4, the units of calculated Al_2O_3 concentration are in milligrams per cubic meter (mg/m^3) and dosage units are in milligram seconds per cubic meter.

If TESTNO (11) above is non-blank, then ISKIP (5) is used only for units selection and the pollutant name is taken from TESTNO (11). Also, calculated precipitation deposition (Model 5) and gravitational deposition (Model 6) are in units of milligrams per square meter.

ISKIP (6) - This option is used for printing purposes only and gives the type of vehicle launch for which calculations are being made and inserts the following in the page heading and plot legend.

If ISKIP (6) is set equal to 1, a "STATIC FIRE" is assumed.

If ISKIP (6) is set equal to 0 or 2, a "NORMAL LAUNCH" is assumed.

If ISKIP (6) is set equal to 4, a "SLOW BURN" is assumed.

If ISKIP (6) is set equal to 5, the program omits this option from the page heading and plot legend.

ISKIP (7) - This option controls the meteorological data used with Model 4.

If ISKIP (7) is set equal to zero, the Program assumes Model 4 is being used to determine concentration, dosage, etc., in a layer where the pollutant distribution at cloud stabilization varies substantially with height. The meteorological data used in Model 4 is automatically determined from the meteorological inputs assigned to the initial layers or sublayers.

If ISKIP (7) is set equal to 1, the Program assumes Model 4 is being used to determine concentration, dosage, etc., resulting from changes in the meteorological layer structure. The meteorological data used in Model 4 after time TAST (time of layer structure change measured from time of cloud stabilization) is taken from the input parameters ALPHL through TEMPL (see page B-22 below).

ISKIP (8) - This option, if set non-zero, prints a detailed listing of all Program inputs.

NSX - Number of radial distances (range) XX in the polar reference grid system. If NXS is set $\leq$ 0, the default value of 41 is used for NXS and the array XX is automatically filled from values shown in Table B-3.

NYS - Number of azimuth bearings in the polar reference grid system. If NYS is set $\leq$ 0, this parameter is automatically calculated and the array of azimuth bearing coordinates (YY) is automatically filled. The value of NYS includes sufficient points in YY to provide a calculation pattern spanning 100 degrees (see Table B-3, note 9).

NZS - Total number of initial layer boundaries including the ground surface boundary.

NCI, NDI, NTI - These parameters each contain two values used in the maximum centerline calculations under ISKIP (2) and in the calculation of isopleths under ISKIP (3).

The total number of isopleth values is given in the hundreds and tens positions of NCI, NDI and/or NTI. If these positions are zero, isopleths for the respective quantity (concentration, dosage, time-mean concentration and/or deposition) is not calculated. The number of critical pollutant levels (air quality standards) to be identified in the plots for maximum centerline calculations is given in the units position of NCI, NDI and/or NTI. If this position is zero, no plot is generated. If set to 9, a plot is generated without indicators for critical pollutant levels (air quality standards).

If the units position of NCI, NDI and/or NTI is greater than zero and not equal to 9, the critical pollutant levels (standards) must be punched as the first values in the arrays CI, DI and/or TI below.

NPTS - Number of heights at which calculations are to be made. If NPTS is set equal to zero or omitted, NPTS is defaulted to 1 and ZZL (1) below is set equal to zero.

NVS - Number of droplet or particle terminal fall velocities used to calculate ground-level gravitational deposition from all layers except the layer in which a destruct occurs (Model 6 only).

NVB - Number of droplet or particle terminal fall velocities used to calculate ground-level gravitational deposition from the layer in which a vehicle destruct occurs (Model 6 only).

XX - Array of radial distances (range) for the coordinates used in calculations on the reference grid system. This array is automatically filled if NXS = 0, (see NXS above). The last 2 points in XX are used only for calculating isopleths; the second to last point should equal 1.2 times the third to the last point and the last point should equal 1.5 times the third to the last point. Space the XX values uniformly and use as many as the program will allow. The user is cautioned to use the default values unless another grid is required.

YY - Array of azimuth bearings for the coordinates used in calculations on the reference grid system measured clockwise from zero degrees north. This array is automatically filled if NYS = 0 (see NYS above). Space the YY values densely toward the center of the calculation sector and use as many as the program will allow. The user is cautioned to use the default values unless another grid is required.

Z - Array of layer boundary heights in ascending order beginning with the surface boundary height (the first layer is always the surface layer).

DELX - Array of the radial distances (range) from the source location (point of cloud stabilization) in each layer to the center of the reference grid system (launch site).

DELY - Array of azimuth bearings to the source location (point of cloud stabilization) in each layer, measured clockwise from zero degrees north.

TABLE B-1

SOURCE STRENGTH INPUT UNITS

Model	Pollutant	
	HCL, CO, CO_2	AL_2O_3
1	1	2
2	1	2
3	1	2
4	1	2
5	2	2
6	2	2

Code definition for Table B-1:

- ① $Q = Q' \frac{22.4}{M} \frac{T}{273.16} \frac{1013.2}{P}$

  (Concentration output units are parts per million (PPM))

- ② $Q = Q'$

  where

  Q = Source strength in each initial layer

  Q' = Total weight of the material in the layer in milligrams

  T = Surface temperature in degrees Kelvin

  P = Surface pressure in millibars

  M = Molecular weight of the material

TABLE B-1 (Continued)

(Deposition output units for Models 5 and 6 are milligrams per square meter (mg/m^2) and concentration output units for Models 1 through 4 are milligrams per cubic meter (mg/m^3))

Q - Source strength within each initial layer. The source strength input units depend upon the model used and the pollutant for which calculations are being made. Table B-1 gives the appropriate input units for each model and pollutant combination.

UBARK - Mean wind speed at ZRK followed by the mean wind speed at the top of each layer.

SIGAK - Standard deviation of the wind azimuth angle for reference time τ_{oK} at ZRK followed by the standard deviation of the wind azimuth angle at the top of each layer.

SIGEK - Standard deviation of the wind elevation angle at ZRK followed by the standard deviation of the wind elevation angle at the top of each layer.

SIGXO - Standard deviation of the alongwind concentration distribution of the source in the layer (alongwind source dimension).

SIGYO - Standard deviation of the crosswind concentration distribution of the source in the layer at a downwind distance XLRY from the true source (crosswind source dimension). The default value is SIGXO.

SIGZO - Standard deviation of the vertical concentration distribution of the source in the layer at a downwind distance XLRZ from the true source (vertical source dimension).

ALPHA - Lateral diffusion coefficient in the layer (default value is 1).

BETA - Vertical diffusion coefficient in the layer (default value is 1).

ZRK - Reference height in the surface layer for meteorological measurements (default value is 2 meters).

TEMPK	-	Virtual potential temperature at each layer boundary z. This parameter is used only in the calculation of the wind speed shear in the layer. If the wind speed shear is negative and the difference between the virtual potential temperature at the top and bottom of the layer is also negative, the Program will use the absolute value of the speed shear. If the temperature difference is positive or zero, the program will use a wind speed shear of zero. If the layer wind speed shear is positive or zero, the virtual potential temperature difference is not used.
TIMAV	-	Time over which time-mean concentration and average cloud concentration are calculated (default value is 600 seconds except for CO, where it is 300 seconds).
THETAK	-	Mean wind direction at ZRK followed by the mean wind direction at the top of each layer.
TAUK	-	Time required for cloud stabilization.
TAUOK	-	Reference time for the standard deviations of the wind azimuth angle SIGAK (default value is 600 seconds).
H	-	Adjusted cloud stabilization height.
XRY	-	Distance downwind from a virtual point source over which recti-linear expansion in the lateral occurs (default value is 100 meters).
XRZ	-	Distance downwind from a virtual point source over which recti-linear expansion in the vertical occurs (default value is 100 meters).
XLRY	-	Reference distance from the true source at which SIGYO is measured (default value is zero).

XLRZ - Reference distance from the true source at which SIGZO is measured (default value is zero).

ZZL - Vertical calculation heights. This parameter can include any heights within the initial layer structure (default value is zero).

IZMOD - This parameter designates the model number or numbers for use in each input layer. A brief description of the six Program models is given below and a complete mathematical description of each model is given in Section 3 of the main body of this report. The possible model number combinations input into IZMOD are given in Table B-2.

1 - Model 1, the source extends vertically through the entire initial layer and turbulent mixing is occurring. It is assumed that the vertical distribution of material is uniform with height and the distributions of material along the along-wind and crosswind cloud axes are Gaussian. The digit 1 is included in the array IZMOD for each layer in which Model 1 is to be used. Also, if any digit of IZMOD is 0, the Program assumes Model 1 has been designated.

2 - Model 2 refers to the same source configuration as Model 1 in that the source extends vertically through the entire depth of the layer and the distribution of material is uniform with height. In Model 2, however, it is assumed that no turbulent mixing is occurring. The digit 2 is included in the array IZMOD for each layer in which Model 2 is to be used in the calculations (IZMOD = 2, 2, 2, etc.).

3 - Model 3 differs from Models 1 and 2 in that the vertical extent of the source is less than the depth of the layer. The model equation thus contains vertical expansion terms. The digit 3 is input to IZMOD for Model 3 (IZMOD = 3).

TABLE B-2

POSSIBLE INPUT MODEL NUMBER COMBINATIONS

IZMOD[1]	PROGRAM ASSUMES CALCULATIONS ARE MADE USING:
0	Model 1
1	Model 1
2	Model 2
3	Model 3
14	Model 1 is used prior to layer transition and Model 4 is used after layer transition occurs at time TAST
24	Model 2 is used prior to layer transition and Model 4 is used after layer transition occurs at time TAST
34	Model 3 is used prior to layer transition and Model 4 is used after layer transition occurs at time TAST
4	Model 4 is used to accomodate to a variation of source strength in the layer and layer transition is immediate (TAST = 0)
5	Model 5 is used and the layer structure and source distribution is assumed to be that of Model 1 when only the digit 5 is given in IZMOD. The digit 5 can be combined with any of the above digit combinations (145, 45, 35, etc.). When a 5 is combined with any of the above digit combinations the Program assumes the layer structure and source distribution of that combination are used with Model 5.
6	Model 6

1 The digits under IZMOD can appear in any order. For example, 14 is the same as 41 and 154 is the same as 415.

4 - Model 4, the layer-transition model, may be used to calculate concentration and dosage resulting from changes in the meteorological layer structure. Model 4 may also be used to calculate concentration and dosage in a layer where the pollutant distribution at cloud stabilization varies substantially with height.

The application of Model 4 requires the following assumptions:

- The boundaries between adjacent initial layers or sublayers is eliminated (at time TAST) and the layers are replaced by a single layer
- Turbulent mixing is occurring in the resultant single layer
- The material in each of the initial layers or sublayers is (before time TAST) uniformly distributed in the vertical
- Reflection occurs at the upper and lower boundaries of the resultant single layer

If the parameters TAST and ISKIP (7) are both set to zero (or omitted from the inputs) and Model 4 is specified for use, the program assumes the function of the model is to accommodate variations in the pollutant distribution with height in the layer at cloud stabilization. For example, the surface mixing layer can be initially divided into several sublayers where the source strength, although assumed to be vertically uniform in each sublayer, varies from layer to layer. In this case the initial layers are immediately reduced to a single layer and Model 4 calculates the contribution from each of the initial sublayers to the composite concentration and dosage field by permitting turbulent mixing across the initial layer boundaries. IZMOD would contain the digit 4 for each of the respective initial sublayers that comprise the resultant single layer.

If Model 4 is to be used to predict the concentration and dosage fields downwind from a change in meteorological structure, the meteorological parameters of the new resultant layer or layers must be specified. Also, the parameter ISKIP (7) must be set equal to 1 and the parameter TAST set equal to the time (after cloud stabilization) at which the layer transition (meteorological structure change) occurs. Each of the initial sublayers that are to be included in a single layer after layer transition are specified by including the digit 4 in the array IZMOD. For example, assume layers 1 through 4 are to be reduced to a single layer after layer transition and layers 5 and 6 are also reduced to a single layer. The first four values of IZMOD would include a 4, but they would also include the number of the model to be used prior to layer transition (14, 24 or 34). The values of IZMOD (5) and (6) for layers 5 and 6 would include a 9 and 4, respectively. The 9 is a special flag to separate the resultant 2 layers after layer transition. Also, these last two values would include the model number to be used prior to layer transition (14, 24 or 34). If Model 1 was to be used with 4 in the above example the IZMOD inputs would be coded as IZMOD = 4, 4, 4, 4, 9, 4 (or IZMOD = 4*4, 9, 4, or IZMOD = 4*14, 19, 14, etc.).

5 - Model 5 is used to calculate the amount of material on the surface by precipitation scavenging. The digit 5 must be included in the array IZMOD for each initial sublayer through which precipitation is occurring. Model 5 uses the layer structure and source distribution defined by any one of Models 1 through 4. Thus, the array IZMOD must include the appropriate model number for each layer that describes the layer structure and source distribution. For example, assume that Model 4 is being used to accommodate to variations in the pollutant distribution with height in the surface mixing layer at cloud stabilization and that the surface mixing layer has been divided into 6 initial sublayers in which the distribution of material can be

considered uniform. Also, assume that precipitation is occurring through all 6 layers. The array IZMOD would then contain six values equal to 45 for each layer 1 through 6 (IZMOD = 6*45).

6 - Model 6 is used to calculate the surface deposition due to gravitational settling. The basic source configuration is a volume source of finite lateral extent and unit vertical extent. Other source configurations are treated by summing the deposition at the ground resulting from a number of basic sources arranged to simulate the desired configuration. The model is essentially a tilted plume model in which the effects of wind shear are taken into account. The axis of a particle or droplet cloud of a given settling velocity intersects the ground plane at a distance from the source and at an angle from the mean surface wind direction that are proportional to the total angular wind shear and the residence time of the settling material in the layers between the source and the ground surface. In any layer, the inclination of the cloud axis from the horizontal is given by $\tan^{-1} V_s / \bar{u}$, where V_s is the particle or droplet settling velocity and $\bar{u}$ is the mean transport wind speed in the layer. In all cases, material released in the K^{th} layer and dispersed upwards by turbulence is assumed to be reflected downward at the interface of the K^{th} and $(K + 1)^{th}$ layers. The basic model is used to calculate the ground-level deposition pattern for a single value of the settling velocity. The total deposition pattern is obtained by summing the results for all settling velocities representative of the particle or droplet-size distribution of the released material on a reference coordinate grid system.

Only IZMOD (1) need be set equal to 6 as no other model can be executed in the same case.

DECAY - Coefficient of time-dependent decay. If DECAY is set > 0, then concentration, dosage, time-mean concentration, etc., are calculated with decay (Does not effect Model 5 or Model 6).

ZLIM - This parameter is the maximum height through which precipitation can occur. If Model 5 is selected, ZLIM is automatically determined from IZMOD. If concentration, dosage, etc., are being calculated with precipitation occurring (BLAMDA > 0.0), ZLIM is equal to the upper boundary of the uppermost layer in which precipitation occurs (ZLIM is defaulted to Z(NZS)).

BLAMDA - Precipitation scavenging (washout) coefficient. If Model 5 is selected, this parameter must be greater than 0. Also, if Model 1, 2, 3 or 4 is selected with BLAMDA > 0 and without Model 5, the Program assumes concentration, dosage, etc., are to be calculated with precipitation occurring.

TIM1 - Time of start of precipitation measured from the time of cloud stabilization.

CI, DI and TI - Arrays of concentration, dosage and time-mean concentration values respectively for which isopleths are calculated. There can be two groups of data in each of these arrays, where both of the groups are arranged in descending order. The values in the first group are critical pollutant levels (air quality standards). The number of values in this group is given in the units position of the parameters NCI, NDI and NTI respectively. The second group of values includes all other isopleth levels desired. The total number of values in CI, DI and TI is given in the hundreds and tens positions of NCI, NDI and NTI respectively. If precipitation, deposition or gravitational deposition is being calculated, the array DI is used for these quantities.

TAST - Time of layer structure change (Model 4) measured from the time of cloud stabilization.

GAMMAP - This parameter is 1 minus the fraction of material reflected at the surface (partial reflection). If this parameter is set to 0, the Program assumes complete reflection; if set equal to .4, 60 percent (.6) reflection is assumed; and, if set equal to 1, no reflection is assumed. If Model 6 is selected and partial reflection is desired, the array GAMMAP must have a value for each particle settling velocity category. For all other models, only GAMMAP (1) need be set.

VS - Droplet or particle terminal fall velocity distribution used in all layers except a layer in which a vehicle destruct occurs (Model 6 only).

PERC - Frequency of occurrence of each velocity category VS (Model 6 only).

ACCUR - Accuracy constant for the line source simulation used in Model 6. A value of 0.45 ensures that the calculated ground deposition is within 10 percent of the deposition expected from a vertical line source. If ACCUR is set to 0.32, the calculated deposition is within 5 percent of that expected from a vertical line source.

VB - Droplet or particle terminal fall velocity distribution used in the layer in which a vehicle destruct occurs. The layer must be the top layer (Model 6 only).

PERCB - Frequency of occurrence of each velocity category VB (Model 6 only).

SCL - Map scale factor in inches for isopleth plots. If the map scale factor is 1 inch = 24000 inches, SCL would be input as 24000. If set to zero, the Program will scale the isopleths within the boundaries defined by XSIZE and YSIZE below.

XMAXIN - Maximum alongwind distance from the launch site in meters for isopleth plots. If set to zero, the Program will use XX(NXS-2) as the maximum distance.

YMAXIN - Maximum crosswind distance for isopleth plots in meters. If set to zero, the Program will calculate YMAXIN.

XSIZE - The number of raster counts on the SC4020 in the X or east-west horizontal plot axis for isopleths. If set to zero, the Program will use 937.

YSIZE - The number of raster counts on the SC4020 in the Y or north-south vertical plot axis for isopleths. If set to zero, the Program will use 899.

RASTIN - The number of raster counts per inch on the SC4020 for isopleth plots. If input as zero, the Program uses 163.2.

XCIZE - The number of raster counts on the SC4020 on the X or alongwind horizontal axis for maximum centerline plots. If set to zero, the Program uses 937.

YCIZE - The number of raster counts on the SC4020 on the vertical axis for maximum centerline plots. If set to zero, the Program uses 899.

XMAXJN - Maximum alongwind distance in meters from the launch site for maximum centerline plots. If set to zero, the Program uses XX(NXS-2).

YMAXJN - Maximum number of log cycles for the vertical axis of the maximum centerline plots if ISW below equals 0 or 2. Maximum value of the vertical axis if ISW below equals 1. If set to zero, the Program determines YMAXJN.

ISW - Maximum centerline plotting flag. If ISW is set to 0 or 2, the Program plots maximum centerline versus distance on a log-log plot. If set to 1, the plot is linear on both axes.

JSW - Isopleth plot switch. If JSW is set equal to 0, the Program will fit a cubic spline function to the discrete isopleth points and plot a smooth curve through the points. If JSW is set equal to 1, the Program will not use the spline function but will plot straight lines between adjacent calculated isopleth points. This option has been included because the spline function sometimes fails to fit the data points when the isopleths are sharply curved. These cases are recognized by a high frequency oscillation along the plotted curve and can be corrected by smoothing the curve by hand or replotting with JSW set equal to 1.

The layer step change (transition) parameters below are used only if ISKIP (7) equals 1 and Model 4 has been selected. These parameters are used only when Model 4 is being used to predict the concentration and dosage downwind from a change in meteorological structure (see IZMOD, Model 4 above).

ALPHL - Lateral diffusion coefficient in each new layer (Default value is 1).

BETL - Vertical diffusion coefficient in each new layer (Default value is 1).

TAUL - Time required for cloud stabilization in the new layers.

TAUOL - Reference time for the standard deviation of the wind azimuth angle SIGAL in the new layers (Default value is 600).

ZRL - Reference height in the surface layer for meteorological measurements. This must be set only if the new bottom layer includes the initial surface layer (Default value is 2).

UBARL - Mean wind speed at the bottom and top boundaries of each new layer. These values are input in ascending order of new layers with the value at the top boundary preceded by the bottom. If the new bottom layer contains the initial surface layer, UBARL at ZRL should be input as the bottom value of this layer.

SIGAL - Standard deviation of the wind azimuth angle for reference time τ_{oL} at the bottom and top boundaries of each new layer. If the new bottom layer contains the initial surface layer, SIGAL at ZRL should be input as the bottom value of this layer.

THETAL - Mean wind direction at the bottom and top boundaries of each new layer. If the new bottom layer contains the initial surface layer, THETAL at ZRL should be input as the bottom value of this layer.

TEMPL - Virtual potential temperature at the bottom and top boundaries of each new layer.

B.3 CONDENSED TABLE OF INPUT PARAMETERS

The data input parameters required for the computer Program are given in condensed form in Table B-3. The information categories in the table are defined as follows:

NAMELIST	-	Name of the FORTRAN NAMELIST list to which the variables belong.
FORTRAN	-	Fortran symbolic notation defining the program input.
MODEL	-	Mathematical notation corresponding to the FORTRAN notation.
UNITS	-	Dimensional units of the input parameters.
LIMITS	-	Numerical limits on input values.
VALUE	-	Default value should the parameter have a value of 0.
ARRAY SIZE	-	Maximum number of core locations for the input parameter.

B.4 DATA INPUT FORMAT

This Program uses the FORTRAN NAMELIST method to input data. Input data must be in a specific form in order to be read using a NAMELIST list. The first character in each card to be read must be blank. The first card in the NAMELIST list contains the NAMELIST name NAM2 preceded by the character $ or &. The last card in the NAMELIST list contains $END (&END) to terminate the list. The form of the remaining data items in the list may be:

a. *Variable Name = Constant* - The *variable name* may be a subscripted array name or a single variable name. Subscripts must be integer constants. The *constant* may be integer, real or Hollerith (nH *alphanumeric characters)* data.

TABLE B-3

TABLE OF INPUT PARAMETERS

NAMELIST	FORTRAN	Model	Units	Limits	Value (3)	Array Size (7)
NAM2	TESTNØ	N/A	N/A	N/A	Blanks	12
	NAMCAS	N/A	N/A	N/A	Blanks	12
	ISKIP	N/A	N/A	(1)	0	15
	NXS	N/A	N/A	≤41	41	1
	NYS	N/A	N/A	≤41	41	1
	NZS	N/A	N/A	≤16	0	1
	NDI	N/A	N/A	≤103 (10)	0	1
	NCI	N/A	N/A	≤103 (10)	0	1
	NTI	N/A	N/A	≤ 103 (10)	0	1
	NPTS	N/A	N/A	≤40	1	1
	NVS	N/A	N/A	≤20	0	1
	NVB	N/A	N/A	≤20	0	1
	XX	R	Meters	> 0.0	(8)	41
	YY	A	Degrees	$0.0 \leq \Phi \leq 360.0$	(9)	41
	NPS	N/A	N/A	0 or 1	0	1
	Z	z_{Bl} and z_{TK}	Meters	≥0.0	z(1) = 0.0	16

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

NAMELIST	FORTRAN	Model	Units	Limits	Value (3)	Array Size (7)
NAM2	DELX	R	Meters	≥ 0.0	0.0	15
	DELY	A	Degrees	0.0 ≤ ϕ ≤ 360.0	0.0	15
	Q	Q	(2)	≥ 0.0	0.0	15
	UBARK	$\bar{u}_R$ and $\bar{u}_{TK}$	Meters Sec^{-1}	≥ 0.1	0.1	16
	SIGAK	$\sigma_{AR}\{\tau_o\}$ & $\sigma_{ATK}\{\tau_o\}$	Degrees	≥ 0.5	0.5	16
	SIGEK	σ_{ER} & σ_{ETK}	Degrees	≥ 0.1	0.1	16
	SIGXØ	$\sigma_{xo}\{K\}$	Meters	> 0.0	N/A	15
	SIGYØ	$\sigma_{yo}\{K\}$	Meters	> 0.0	SIGXØ	15
	SIGZØ	$\sigma_{zo}\{K\}$	Meters	≥ 0.0	0.0	15
	ALPHA	α_K	N/A	≥ 0.0	1.0	15
	BETA	β_K	N/A	≥ 0.0	1.0	15
	ZRK	z_R	Meters	≥ z(1)	2.0	1
	TIMAV	T_A	Seconds	≥ 0.0	600 or 360	1
	THETAK	θ_{B1} & $\dot{\theta}_{TK}$	Degrees	$0.0 \le \theta_K \le 360.0$	0.0	16

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

NAMELIST	FORTRAN	Model	Units	Limits	Value (3)	Array Size (7)
NAM2	TAUK	τ	Seconds	>0.0	N/A	1
	TAUØK	τ_o	Seconds	≥ 0.0	600.0	1
	H	H	Meters	≥ 0.0	0.0	1
	XRY	x_{ry}	Meters	≥ 0.0	100.0	1
	XRZ	x_{rz}	Meters	≥ 0.0	100.0	1
	XLRY	x_{Ry}	Meters	≥ 0.0	0.0	1
	XLRZ	x_{Rz}	Meters	≥ 0.0	0.0	1
	ZZL	z	Meters	≥ 0.0	0.0	1
	IZMØD	N/A	N/A	(11)	.1	15
	DECAY	k	Seconds^{-1}	≥ 0.0	0.0	1
	ZLIM	z_{lim}	Meters	$= z_{TK}$	Z (NZS) (13)	1
	TIMI	t_1	Seconds	≥ 0.0	(5)	1
	BLAMDA	Λ	Seconds^{-1}	≥ 0.0	(5)	1
	DI	$D_K \{x_K, y_K, z_K\}$	(4)	≥ 0.0	(5)	10
	CI	$\chi_K \{x_K, y_K, z_K\}$	(4)	≥ 0.0	(5)	10
	TI	$\chi_K \{x_K, y_K, z_K; T_A\}$	(4)	≥ 0.0	(5)	10

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

NAMELIST	FORTRAN	Model	Units	Limits	Value (3)	Array Size (7)
NAM2	TAST	t^*	Seconds	≥0.0	0.0	5
	TEMPK	Φ_{BI} & Φ_{TK}	Degrees K	≥0.0	0.0	16
	VS	V_s	Meters sec^{-1}	≥0.0	(5)	20
	PERC	f_i	N/A	>0.0	(5)	20
	ACCUR	R_c	N/A	(6)	(5)	20
	VB	V_{SK}	Meters Sec^{-1}	>0.0	(5)	20
	PERCB	f_i	N/A	>0.0	(5)	20
	HB	H_{SK}	Meters	≥0.0	0.0	1
	GAMMAP	$1-\gamma_r$	N/A	≥0 & ≤1	0.0	20
	ALPHL	α_L	N/A	≥0.0	(13)	5
	BETL	β_L	N/A	≥0.0	(13)	5
	TAUL	τ	Seconds	>0.0	TAUK	1
	TAUØL	τ_o	Seconds	≥0.0	TAUØK	1
	ZRL	z_{RL}	Meters	≥2.0	ZRK	1
	UBARL	$\bar{u}_{BL}$ & $\bar{u}_{TL}$	Meters Sec^{-1}	≥0.0	(12)	10
	SIGAL	$\sigma_{ABL}\{\tau_o\}$ & $\sigma_{ATL}\{\tau_o\}$	Degrees	≥0.0	(12)	10

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

NAMELIST	FORTRAN	Model	Units	Limits	Value (3)	Array Size (7)
NAM2	SIGEL	σ_{EBL} & σ_{ETL}	Degrees	≥0.0	(12)	10
	THETAL	θ_{BL} & θ_{TL}	Degrees	≥0.0 & ≤360.0	(12)	10
	TEMPL	Φ_{BL} & Φ_{TL}	Degrees K	≥0.0	0.0	10
	SCL	N/A	Inches	≥0	Calculated	1
	XMAXIN	R	Meters	≥0	Calculated	1
	YMAXIN	N/A	Meters	≥0	Calculated	1
	XSIZE	N/A	Rasters	≥0	937	1
	YSIZE	N/A	Rasters	≥0	899	1
	RASTIN	N/A	Rasters/ Inch	≥0	163.2	1
	XCIZE	N/A	Rasters	≥0	937	1
	YCIZE	N/A	Rasters	≥0	899	1
	XMAXJN	N/A	Meters	≥0	XX(NXS-2)	1
	YMAXJN	N/A	Log Cycles or Meters	≥0	Calculated	1
	ISW	N/A	N/A	1 or 2	2	1
	JSW	N/A	N/A	0 or 1	0	1

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

(1) See Section B-2 for the range of values of the ISKIP options.

(2) Units depend on model; see Section B-2 in the definition of Q.

(3) The column under Value is used to simplify the Program input deck by providing default values should the parameter be intentionally omitted in the first data case or set to zero. All parameters in Table B-3 remain their previous value for all subsequent cases unless changed in the input list.

(4) Units of dosage and concentration isopleth values must be consistent with Program output units, milligrams/meter^3 or parts per million, etc.

(5) These parameters must have values other than zero only if they are used by the model selected and only in the applicable layers.

(6) See Section B-2 for the description of ACCUR.

(7) Several variables are dimensioned to a larger value in the Program, but the extra space is used for other purposes.

TABLE B-3

TABLE OF INPUT PARAMETERS

(Continued)

(8) The default values of XX are: 500, 1250, 2500, 3750, 5000, 6250, 7500, 8750, 10000, 11250, 12500, 13750, 15000, 16250, 17500, 18750, 20000, 21250, 22500, 23750, 25000, 26250, 27500, 28750, 30000, 31250, 32500, 33750, 35000, 36250, 37500, 38750, 40000, 41250, 42500, 43750, 45000, 47500, 50000, 65000, 80000 meters. Default values of XX are used only if NXS is set to 0.

(9) The default values of the YY are the average layer wind direction $\pm 180^{o}$ rounded to the nearest 5^{o} added to each of the following angles: -40, -35, -30, -27, -24, -22, -20, -18, -16, -14, -12, -10, -8, -7, -6, -5, -5, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 14, 16, 18, 20, 22, 24, 27, 30, 40 degrees.

(10) The limit values given for NDI, NCI and NTI mean there is a maximum of 10 possible isopleth values with a maximum of 3 critical pollutant levels (air quality standards) within the 10. The total number of values is input in the tens and hundreds positions and the number of critical pollutant levels is input in the units position.

(11) IZMØD is a 3 digit integer where any one of the three digits can be an integer from 0 to 6 or the integer 9. See Section B.2 for a complete explanation of IZMØD.

(12) If these parameters are input, both bottom and top values are input respectively for each new layer in the layer step change.

(13) ZLIM is automatically calculated if IZMØD contains a 5 (Model 5).

b. *Array Name = Set of Constants* (separated by commas) - The *array name* is not subscripted. The *set of constants* consists of constants of the type integer or real. The number of constants must be less than or equal to the array size. Successive occurrences of the same constant can be represented in the form *k* constant*.

The sequence of the input data parameters within the list is not significant. A more detailed explanation of the FORTRAN NAMELIST can be found in most FORTRAN language manuals. Section B.5 shows an example input data coding sheet. All Program input parameters are set to zero prior to input of the first case. Parameters that are not used or have default values need not appear in the input deck. When multiple cases are stacked, all parameters retain their values from the last case and are changed only by input.

B.5 SAMPLE PROBLEM INPUT DATA

The sample problem is for the hypothetical launch of a Space Shuttle vehicle. The meteorological parameters were measured at Kennedy Space Center for the 21 October 1972 case described in Section 5. The parameter input sheet is shown in Figure B-1 and the Program output is shown in Appendix C.2. The data input parameters shown in Figure B-1 were automatically calculated by the Preprocessor Program and are shown in the Preprocessor output in Appendix D.1. Only the HCl normal launch cases with Model 3 and Model 4 are given here. The first namelist deck shown in Figure B-1 is for Model 4 and the second is for Model 3.

B.6 LINKAGE DIAGRAM OF THE NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM - VERSION 5

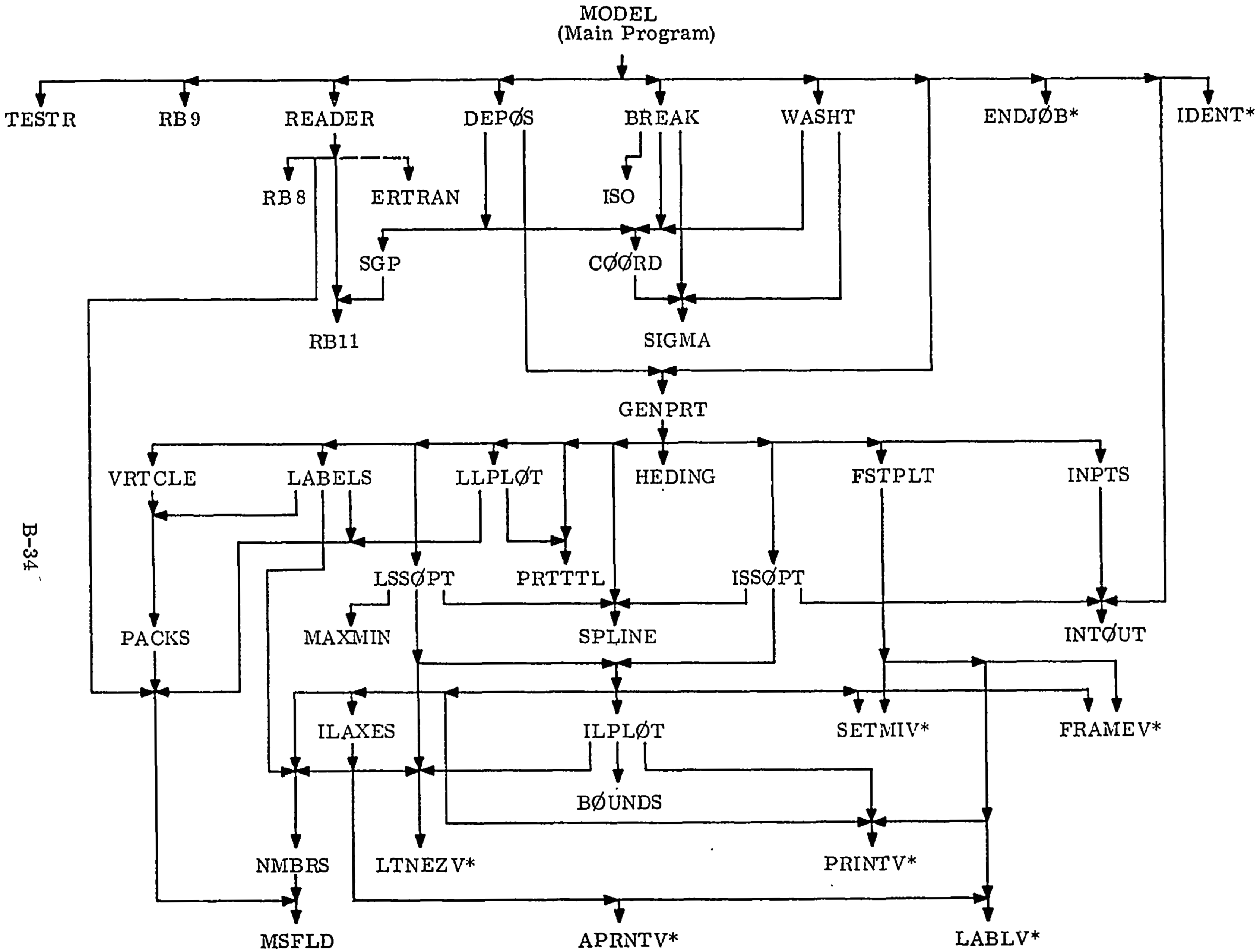
Figure B-2 shows the subroutine linkage diagram for the NASA/MSFC Multilayer Diffusion Models Program - Version 5. Each line terminating at a subroutine name represents a subroutine call. The asterisk indicates subroutines in the SC4020 plot package used only in the UNIVAC 1108 program copy at Marshall

```
$NAM2
TESTNØ=60HKSC 21 OCT 72.                          SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,1,2,NPS=0,NZS=13,NDI=69,NCI=69,NTI=62,ZRK=18,TAUK=447.273,
IZMØD=12*4,H=1790,TIMAV=600,
Z=0,194,250,284,500,558,637,750,1000,1098,1135,1250,1432,
Q=4.422763E7,1.7876125E7,1.3832882E7,1.7371957E8,8.5352765E7,1.5651788E8,
3.3045618E8,1.4067362E9,9.0119715E8,4.0207076E8,1.4819777E9,3.0778224E9,
UBARK=6,8,9,10,10,10,10,6*11,SIGAK=13*4.5,SIGEK=13*4.5,
SIGXØ=12*532.837,SIGYØ=12*532.837,SIGZØ=12*0,
THETAK=80,81,2*82,2*79,78,76,71,68,67,63,56,DI=400,200,100,50,25,5,
CI=16,8,4,1,.5,.1,TI=30,4,8,2,1,.5,
DELX=39.922,67.179,88.489,279.077,343.023,438.965,601.038,1053.551,1259.946,
1340.737,1610.334,2083.997,
DELX=260.5,260.906,261.169,260.712,260.393,259.979,259.175,256.734,255.544,
255.057,253.36,250.167,
TEMPK=295.624,297.076,297.521,297.781,297.137,296.876,296.979,297.192,297.393,
297.523,297.445,297.538,297.58,
$END
```

FIGURE B-1. Sample problem input data generated by the Preprocessor Program for the launch of a Space Shuttle Vehicle.

```
$NAM2
TESTNØ=60HKSC 21 OCT 72.                        SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,1,2,NPS=1,NZS=2,NDI=69,NCI=69,NTI=62,ZRK=18,TAUK=447.273,IZMØD=3,
Z=0,1432,Q=8.091787E9,UBARK=6,11,SIGAK=2*4.5,SIGEK=2*4.5,SIGXØ=532.837,
SIGYØ=532.837,SIGZØ=183.163,THETAK=80,56,TIMAV=600,DELX=4103.229,DELY=235.455,
DI=400,200,100,50,25,5,CI=16,8,4,1,.5,.1,TI=30,4,8,2,1,.5,H=1038.2,
TEMPK=295.624,297.076
$END
```

FIGURE B-1 (Continued) Sample problem input data generated by the Preprocessor Program for the launch of a Space Shuttle Vehicl



FIBURE B-2. Linkage diagram of the subroutines in the NASA/MSFC Multilayer Diffusion Models Computer Program - Version 5.

Space Flight Center, Huntsville, Alabama. The program also references the FORTRAN library functions SIN, COS, ATAN2, EXP, SQRT, ALOG and ALOG10 not shown in Figure B-2. A description of the routines shown in Figure B-2 is given in Section B.8.

B.7 DESCRIPTION OF PROGRAM SUBROUTINES

Subroutine MODEL is the main controlling program in the NASA/MSFC Multilayer Diffusion Models Program - Version 5. This routine controls the calculations for Models 1 through 4 and passes control to subroutines that calculate Models 5 and 6.

Subroutine BREAK completes all calculations for Models 1 through 4 (Section 3.1 to 3.4).

Subroutine READER is the data input routine for the Program. All input default values are set in this routine and most meteorological layer parameters are calculated in this routine.

Subroutine DEPOS controls all calculations for Model 6 (Section 3.6).

Subroutine SGP completes all calculations for Model 6 (Section 3.6).

Subroutine WASHT completes all calculations for Model 5 (Section 3.5).

Subroutine TESTR is used to determine special indices used in calculations with MODEL 4. This routine also calculates the downwind distance at which the layer transition for Model 4 begins.

Subroutine IDENT is used to initialize the SC4020 plotting routines used at Marshall Space Flight Center.

Subroutine ENDJOB is used to close out the SC4020 plotting routines at the end of a job.

Subroutine RB8 is used in the calculation of the wind speed and the standard deviation of the elevation and the azimuth wind angles in the surface layer (q, m, p, ql, ml and pl in Sections 3.1 to 3.4).

Subroutine ERTRAN is a time and date routine. This routine is a UNIVAC system utility routine that returns the data and time. The following statement is used to retrieve the 6 character date (MMDDYY) in the first variable and the 6 character time (HHMMSS) in the second variable:

```
CALL ERTRAN(9,NTFB(1), NTFB(2))
```

Subroutine ISO evaluates the normal error function used in the calculation of Model 4 (Equation 3-18, Section 3.4).

Subroutine COORD calculates the downwind and crosswind distances of a grid system calculation point relative to the alongwind cloud axis and the layer source location.

Subroutine RB11 is used in the calculation of the wind speed and the standard deviation of the elevation and azimuth wind angle in the surface layer. (Equations (3-2), (3-5), (3-17), (3-19), (3-23), and (3-27), Section 3.1 to 3.4).

Subroutine SIGMA calculates the crosswind, alongwind and vertical standard deviations of the dosage distribution for Models 1 through 5.

Subroutine GENPRT controls the printing and/or plotting of all model calculations.

Subrouting LABELS generates the page heading information for all calculations.

Subroutine ISSOPT plots the isopleths of concentration, dosage, deposition, and/or time-mean concentration on the SC4020 plotter.

Subroutine LLPLOT plots maximum concentration, dosage, deposition, and/or time-mean concentration as a function of distance in the form of a printer plot on the UNIVAC 1108.

Subroutine PRTTTL prints all page heading information.

Subroutine NMBRS converts a real number to a sequence of BCD characters for insertion into the page heading and for plotting numeric values.

Subroutine LSSOPT plots maximum centerline concentration, dosage, deposition, and/or time-mean concentration as a function of distance on the SC4020 plotter.

Subroutine PACKS removes extra blank characters from the page heading and packs it into a form suitable for printing.

Subroutine MSFLD extracts a bit string (byte) from one word and stores it into another.

Subroutine HEDING sets up key flags that tell the LABELS subroutine whether the label is concentration, dosage, label or other.

Subroutine INPTS is used to set indices for reading or writing records from mass storage.

Subroutine INTOUT writes or reads records from mass storage.

Subroutine VRTCLE determines the vertical axis label for maximum centerline plots.

Subroutine FSTPLT plots information identifying the plots for a set of input data decks.

Subroutine ILAXES draws and labels the plot grids for maximum centerline and isopleth plots.

Subroutine ILPLOT draws and labels the maximum centerline curve and the isopleth curve.

Subroutine MAXMIN determines the maximum and minimum plot values for maximum centerline plots if they are not input.

Subroutine BOUNDS determines whether the plot curve leaves or enters the plot boundaries and interpolates for the intersection of the curve and boundary.

Subroutine SPLINE calculates the coefficients of a third-degree natural spline function used to calculate maximum centerline values and to plot maximum centerline and isopleth curves. The function derives its name from the fact that such a curve approximates the behavior of a mechanical spline used by draftsman to draw a smooth curve through a set of points. The curve has the properties of having continuous first and second derivatives, and of being the "smoothest" curve through a given set of points in the sense that it minimizes

$$\sigma_K = \int_{x_1}^{x_2} \left[s^k (x)\right]^2 dx$$

where

s	=	dosage, concentration, etc.
x	=	azimuth bearing or distance.
k	=	derivative = 1, 2

The function s(x) is the unique curve with these properties and is a piece wise function given by a polynomial of maximum degree of 3 in each of the intervals $[x_i, x_i + 1)$, in general by a different polynomial in each such interval. For a more detailed explanation of the spline function, the reader is referred to the book by T. N. E. Greville, Theory and Applications of Spline Functions.

Subroutine LINE2V draws a line between 2 points (SC4020).

Subroutine PRINTV prints alphanumeric information horizontally (SC4020).

Subroutine FRANIEV advances the camera frame (SC4020).

Subroutine LABLV prints a number (SC4020).

Subroutine APRNTV prints alphanumeric information vertically (SC4020).

Subroutine SETMIV sets the plot margins (SC4020).

APPENDIX C

FORTRAN SOURCE LISTINGS FOR THE PREPROCESSOR PROGRAM AND THE NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM - VERSION 5

C.1 FORTRAN SOURCE LISTING FOR THE PREPROCESSOR PROGRAM

This section contains the complete FORTRAN source listing of the Preprocessor Program.

```
C     * PLUME RISE AND SOURCE DISTRIBUTION PREPROCESSOR PROGRAM FOR USE   VHC00100
C     * WITH THE NASA/MSFC MULTILAYER MODEL VERSION V.                    VHC00200
C     *                                                                   VHC00300
C                                                                         VHC00400
C       *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-VHC00500
C                                                                         VHC00600
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(VHC00700
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL               VHC00800
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                            VHC00900
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS             VHC01000
      COMMON /SIG/ SIGXO(20),SIGYO(20),SIGZO(20),H                        VHC01100
      COMMON /F/ DATE(6),FRQ(4),WTMOL(3)                                  VHC01200
      DIMENSION IPOL(4),I1(160),I2(12),I3(61),I4(13),I5(142),ITP(2),JP(2VHC01300
     1),ISW(7)                                                            VHC01400
      DIMENSION WS(21),I6(44)                                             VHC01500
      INTEGER YES,VEHICL,TYPE,TYPES,UNITS                                 VHC01600
      DIMENSION TYPE(8),QC1(8),QT1(8),QC2(8),QT2(8),QC3(8),QT3(8),AA(8),VHC01700
     1BB(8),HEATN(8),HEATA(8),FRQ1(4,8),FRQ2(4),TYPES(12),UNITS(4),      VHC01800
     2LNCH(11),LNTL(8),HEATM(8)                                           VHC01900
      DIMENSION DDI(10,4),CCI(10,4),TTI(10,4)                             VHC02000
      DATA TYPE/3HTTN,3HSTL,3HDTH,3HMIN,4*0/                              VHC02100
      DATA QC1/4.174374E6,9.384984E6,9.18E5,3.7708565E5,4*0./             VHC02200
      DATA QT1/4.67529888E8,9.1562698E8,6*0./                             VHC02300
      DATA QC2/1.742E6,3.75257E6,6*0./                                    VHC02400
      DATA QT2/1.95104E8,4.5781349E8,6*0./                                VHC02500
      DATA QC3/1.301E6,3.05208994E6,2.958E5,9.567785E4,4*0./              VHC02600
      DATA QT3/3.903E8,9.1562698E8,8.8741E7,2.8703354E7,4*0./             VHC02700
      DATA AA/.63463,.663552,1.32095,.439907,4*0./                        VHC02800
      DATA BB/.4837,.485477,.39457,.47879,4*0./                           VHC02900
      DATA HEATN/2500.,2582.,625.88,691.0,4*0.0/                          VHC03000
      DATA HEATM/1036.,1274.,6*0./                                        VHC03100
      DATA HEATA/4*1000.0,4*0./                                           VHC03200
C     FRACTIONAL DIST FOR MINUTEMAN II ABNORMAL LAUNCH                    VHC03300
      DATA FRQ2/.2042109,.2188377,.0,.2799764/                            VHC03400
      DATA FRQ1/.21,.279,.029,.304,.207,.28,.0,.304,.137278,.307976,.0, VHC03500
     1.24892,.1973,.22046,.0,.27683,16*0./                                VHC03600
      DATA TYPES/72HTITAN IIIC          SPACE SHUTTLE     DELTA-THOR      VHC03700
     1  MINUTEMAN II      /                                               VHC03800
      DATA UNITS/24H PPM          ML/M**3    /                            VHC03900
      DATA LNCH/66H     NORMAL LAUNCHSINGLE ENGINE BURN         SLOW BURVHC04000
```

```
      1NVEHICLE     /                                                   VHC04100
       DATA NAMT/12*1H /                                                VHC04200
       DATA CCI/16.0,8.0,4.0,1.0,0.5,0.1,4*0.0,35.0,10.0,4.0,2.0,1.0,0.1,VHC04300
      14*0.0,20.0,10.0,5.0,3.0,1.0,0.5,4*0.0,2.0,1.0,0.4,0.1,0.05,0.01,4*VHC04400
      20.0/                                                             VHC04500
       DATA DDI/400.0,200.0,100.0,50.0,25.0,5.0,4*0.0,400.0,200.0,100.0, VHC04600
      150.0,25.0,5.0,4*0.0,400.0,200.0,100.0,50.0,25.0,5.0,4*0.0,40.0,   VHC04700
      220.0,10.0,5.0,2.5,0.5,4*0.0/                                     VHC04800
       DATA TTI/30.0,4.0,8.0,2.0,1.0,0.5,4*0.0,150.0,100.0,60.0,30.0,15.0VHC04900
      1,1.0,4*0.0,5.0,2.0,1.0,.5,.1,.05,4*0.0,50.0,100.0,25.0,10.0,5.0,1.VHC05000
      20,4*0.0/                                                         VHC05100
       DATA JP/12H FEET METERS/,NO/3H NO/,NO1/3HNO1/,NO2/3HNO2/         VHC05200
       DATA YES/3HYES/,IPOL/24H HCL    CO    CO2   AL203/,ITP/1HF,1HK/   VHC05300
       DATA IAST/1H*/                                                   VHC05400
       DATA IBLK/3H   /,IBNK/1H /,NMS/1HM/                              VHC05500
       COMMON /OUT/ GF(20,4),WD(21),ISKIP(10),NDI,NCI,NTI,ZRK,JBOT,JTOP, VHC05600
      1IZMOD(21),CI(10),DI(10),TI(10),NPS                               VHC05700
       COMMON /DISPL/ DXX,DX(21),DYY,DY(21),ILXY                        VHC05800
       EQUIVALENCE (WS,UBAR),(I1,QC),(I2,ZM),(I3,SIGXO),                VHC05900
      1(I4,DATE),(I5,QF),(I6,DXX)                                       VHC06000
C                                                                       VHC06100
C                   *** PROGRAM INPUTS ***                              VHC06200
C      NAMCAS - GENERAL DATA SET TITLING INFORMATION (CARD 1 COL 1-72)  VHC06300
C               IF INPUT AS BLANKS THE INFORMATION IN THE LAST CASE INPUVHC06400
C               IS USEL                                                 VHC06500
C      VEHICL - THREE CHARACTERS GIVING THE VEHICLE TYPE (CARD 2 COL 2-4)VHC06600
C                TTN IS TITAN IIIC VEHICLE                              VHC06700
C                STL IS SPACE SHUTTLE VEHICLE                           VHC06800
C                DTH IS DELTA-THOR VEHICLE                              VHC06900
C                MIN IS MINUTEMAN II VEHICLE                            VHC07000
C      NORMAL - THREE CHARACTERS GIVING THE TYPE OF LAUNCH(CRD 2 COL 5-7)VHC07100
C                YES IS A NORMAL LAUNCH                                 VHC07200
C                NO1 IS A SINGLE ENGINE BURN ABNORMAL LAUNCH            VHC07300
C                NO2 IS A SLOW BURN ABNORMAL LAUNCH                     VHC07400
C      IFLET - 1 CHARACTER IF Z IS IN FEET PUNCH F, IF Z IS IN METERS   VHC07500
C               PUMCH M, (CARD 2 COL 8)                                 VHC07600
C      KNOTS - 1 CHARACTER IF WS IS IN METERS/SEC PUNCH M, IF WS IS IN  VHC07700
C               KNOTS PUNCH K. (CARD 2 COL 9)                           VHC07800
C      DATE - 36 CHARACTERS IDENTIFYING THE METEOROLOGICAL DATA CASE    VHC07900
C             WITHIN THE GENERAL DATA CASE IDENTIFIED IN NAMCAS ABOVE   VHC08000
C             (CARD 2 COL 10-45)                                        VHC08100
```

```
C     SIGAR - STANDARD DEVIATION OF THE WIND AZIMUTH ANGLE AT THE        VHC08200
C              SURFACE MEASUREMENT HEIGHT (DEGREES) (CARD 2 COL 46-55)    VHC08300
C              F10.0 FORMAT )                                             VHC08400
C     RHO - SURFACE AIR DENSITY (G/M**3) (CARD 2 COL 56-65 F10.0 FORMAT)VHC08500
C     ISW(1) - IF SET TO 1 CALCULATE PARAMETERS FOR MODEL 4              VHC08600
C               IF SET TO 0 MODEL 4 IS NOT PRODUCED (CARD 2 COL 67)      VHC08700
C     ISW(2) - IF SET TO 1 CALCULATE PARAMETERS FOR MODEL 3              VHC08800
C               IF SET TO 0 MODEL 3 IS NOT PRODUCED (CARD 2 COL 68)      VHC08900
C     ISW(3) - IF SET TO 1 DATA FOR HCL IS PRODUCED                      VHC09000
C               IF SET TO 0 HCL ISNOT PRODUCED (CARD 2 COL 69)           VHC09100
C      ISW(4) - IF SET TO 1 DATA FOR CO IS PRODUCED                      VHC09200
C               IF SET TO 0 CO IS NOT PRODUCED (CARD 2 COL 70)           VHC09300
C     ISW(5) - IF SET TO 1 DATA FOR AL203 IS PRODUCED                    VHC09400
C               IF SET TO 0 AL203 IS NOT PRODUCED (CARD 2 COL 71)        VHC09500
C     ISW(6) - IF SET TO 1 DATA FOR CO2 IS PRODUCED                      VHC09600
C               IF SET TO 0 CO2 IS NOT PRODUCED (CARD 2 COL 72)          VHC09700
C     ISW(7) - IF SET TO 1 THE CLOUD TRAJECTORY COORDINATES DELX,DELY    VHC09800
C               ARE CALCULATED AND PUNCHED FOR EACH LAYER.  IF SET TO 0  VHC09900
C               CLOUD TRAJECTORY COORDINATES ARE NOT CALCULATED          VHC10000
C               (CARD 2 COL 73)                                          VHC10100
C     THE FOLLOWING PARAMETERS EXCEPT IHM ALL USE AN F10.0 FORMAT        VHC10200
C     Z - HEIGHT OF LAYER BOUNDARIES (FEET OR METERS) COL 1-10 CARDS 3-NVHC10300
C     WD - WIND DIRECTION AT EACH Z (DEGREES) COL 11-20 CARDS 3-N        VHC10400
C     WS - WIND SPEED AT EACH Z (KNOTS OR METERS/SEC) COL 21-30 CARDS3-NVHC10500
C     T - TEMPERATURE AT EACH Z (DEGREES C) COL 31-40 CARDS 3-N          VHC10600
C     P - PRESSURE AT EACH Z (MILLIBARS) COL 41-50 CARDS 3-N             VHC10700
C     RH - RELATIVE HUMIDITY AT EACH Z (PERCENT) COL 51-60 CARDS 3-N     VHC10800
C     IHM - ASTERISK (*) IN COLUMN 80 IF THE HEIGHT Z ON THIS CARD IS    VHC10900
C     THE SURFACE MIXING LAYER HEIGHT HM.  IF NOT FOUND THE LAST Z INPUTVHC11000
C     IS USED FOR HM.                                                    VHC11100
C                                                                        VHC11200
C     INITIALIZE CORE TO ZERO                                            VHC11300
    5 DO 10 I=1,160                                                      VHC11400
      I1(I) = 0                                                          VHC11500
      IF (I .GT. 142) GO TO 10                                           VHC11600
      I5(I) = 0                                                          VHC11700
      IF (I .GT. 61) GO TO 10                                            VHC11800
      I3(I) = 0                                                          VHC11900
      IF (I .GT. 44) GO TO 10                                            VHC12000
      I6(I) = 0                                                          VHC12100
      IF (I .GT. 12) GO TO 10                                            VHC12200
```

```
      I2(I) = 0                                                         VHC12300
      IF (I .GT. 13) GO TO 10                                           VHC12400
      I4(I) = 0                                                         VHC12500
   10 CONTINUE                                                          VHC12600
C                                                                       VHC12700
C                    *** PROGRAM CONSTANTS ***                          VHC12800
C     QC1-TOTAL SOURCE OUTPUT RATE IN GRAMS/SEC FOR A NORMAL LAUNCH     VHC12900
C     QT1- TOTAL SOURCE STRENGTH IN GRAMS FOR NORMAL LAUNCH             VHC13000
C     QC2 - TOTAL SOURCE OUTPUT RATE IN GRAMS/SEC FOR AN ABNORMAL LAUNCHVHC13100
C           WITH ONE ENGINE BURNING ON PAD                              VHC13200
C     QT2 - TOTAL SOURCE STRENGTH IN GRAMS FOR AN ABNORMAL LAUNCH WITH  VHC13300
C           ONE ENGINE BURNING ON PAD                                   VHC13400
C     QC3 - TOTAL SOURCE OUTPUT RATE IN GRAMS/SEC FOR AN ABNORMAL LAUNCHVHC13500
C           WHERE ENGINES EXPLODE AND BURN ON GROUND                    VHC13600
C     QT3 - TOTAL SOURCE STRENGTH IN GRAMS FOR AN ABNORMAL LAUNCH WHERE VHC13700
C           THE ENGINES EXPLODE AND BURN ON GROUND                      VHC13800
C     AA AND BB - ROCKET RISE PARAMETERS IN EQUATION TR=AA*Z**BB        VHC13900
C     HEATN - HEAT OUTPUT (CAL/G) NORMAL LAUNCH                         VHC14000
C     HEATM - HEAT OUTPUT (CAL/G) ABNORMAL LAUNCH WITH SINGLE ENGINE    VHC14100
C             BURN                                                      VHC14200
C     HEATA - HEAT OUTPUT (CAL/G) ABNORMAL LAUNCH WITH SLOW BURN ON PAD VHC14300
C     GAMMAI - ENTRAINMENT PARAMETER FOR NORMAL LAUNCH                  VHC14400
      GAMMAI = 0.64                                                     VHC14500
C     GAMMAC - ENTRAINMENT PARAMETER FOR ABNORMAL LAUNCH                VHC14600
      GAMMAC = 0.5                                                      VHC14700
C     FRQ - FRACTIONAL DISTRIBUTION OF MATERIAL FOR HCL, CO, CO2, AL203 VHC14800
C     WTMOL - MOLECULAR WEIGHTS OF HCL, CO, CO2                         VHC14900
      WTMOL(1) = 36.46                                                  VHC15000
      WTMOL(2) = 28.01                                                  VHC15100
      WTMOL(3) = 44.01                                                  VHC15200
C     G - ACCELERATION OF GRAVITY (M/SEC SQUARE)                        VHC15300
      G = 9.8                                                           VHC15400
C     CP - SPECIFIC HEAT OF AIR                                         VHC15500
      CP = 0.24                                                         VHC15600
C     PI - RADIANS IN 180 DEGREES                                       VHC15700
      PI = 3.1415926                                                    VHC15800
      ND1 = 69                                                          VHC15900
      NC1 = 69                                                          VHC16000
      ISKIP(1) = 0                                                      VHC16100
      ISKIP(2) = 3                                                      VHC16200
      ISKIP(3) = 3                                                      VHC16300
```

```
      NPS = 0                                                             VHC16400
      READ 1002, NAMCAS                                                   VHC16500
      DO 15 I=1,12                                                        VHC16600
      IF (NAMCAS(I) .NE. IBLK) GO TO 17                                   VHC16700
   15 CONTINUE                                                            VHC16800
      DO 16 I=1,12                                                        VHC16900
   16 NAMCAS(I) = NAMT(I)                                                 VHC17000
   17 DO 18 I=1,12                                                        VHC17100
   18 NAMT(I) = NAMCAS(I)                                                 VHC17200
      READ 1000, VEHICL,NORMAL,IFEET,KNOTS,DATE,SIGAR,RHO,ISW             VHC17300
      SIGAP(1) = 0.5*SIGAR                                                VHC17400
      K = 1                                                               VHC17500
   20 READ 1001, Z(K),WD(K),WS(K),T(K),P(K),RH(K),IHM                     VHC17600
      IF (IHM .EQ. IAST) HM = Z(K)                                        VHC17700
      IF (Z(K)+WS(K)) 21,22,21                                            VHC17800
   21 K = K+1                                                             VHC17900
      GO TO 20                                                            VHC18000
   22 NZS = K-1                                                           VHC18100
      IPNPS = 4                                                           VHC18200
      IF (ISW(5) .GT. 0) GO TO 45                                         VHC18300
      IPNPS = 3                                                           VHC18400
      IF (ISW(6) .GT. 0) GO TO 45                                         VHC18500
      IPNPS = 2                                                           VHC18600
      IF (ISW(4) .GT. 0) GO TO 45                                         VHC18700
      IPNPS = 1                                                           VHC18800
   45 CONTINUE                                                            VHC18900
C     ZRK - HEIGHT AT WHICH SIGAR IS MEASURED (METERS)                    VHC19000
      ZRK = Z(1)                                                          VHC19100
      IF (NORMAL .EQ. IBLK) NORMAL = YES                                  VHC19200
      IF (VEHICL .EQ. TYPE(1)) GO TO 50                                   VHC19300
      IF (VEHICL .EQ. TYPE(2)) GO TO 51                                   VHC19400
      IF (VEHICL .EQ. TYPE(3)) GO TO 52                                   VHC19500
      IF (VEHICL .EQ. TYPE(4)) GO TO 53                                   VHC19600
      PRINT 2009                                                          VHC19700
   50 JV = 1                                                              VHC19800
      GO TO 60                                                            VHC19900
   51 JV = 2                                                              VHC20000
      ISW(6) = 0                                                          VHC20100
      GO TO 60                                                            VHC20200
   52 JV = 3                                                              VHC20300
      ISW(6) = 0                                                          VHC20400
```

```
      IF (NORMAL .EQ. NO1) NORMAL = NO2
      GO TO 60
   53 JV = 4
      ISW(6) = 0
      IF (NORMAL .EQ. NO1) NORMAL = NO2
   60 IF (JV .NE. 4) GO TO 61
      IF (NORMAL .NE. YES) GO TO 63
   61 DO 62 I=1,4
   62 FRQ(I) = FRQ1(I,JV)
      GO TO 65
   63 DO 64 I=1,4
   64 FRQ(I) = FRQ2(I)
   65 CONTINUE
      IF (IFEET .EQ. IBNK) IFEET = NMS
      IF (KNOTS .EQ. IBNK) KNOTS = NMS
      IF (NORMAL .EQ. YES) GO TO 32
      IF (NORMAL .EQ. NO1) GO TO 31
      IF (NORMAL .EQ. NO2) GO TO 30
      PRINT 2006, NORMAL
      GO TO 500
   30 QC = QC3(JV)
      QT = QT3(JV)
      IJM = 7
      GAMMA = GAMMAC
      HEAT = HEATA(JV)
      NORMAL = 0
      ISKIP(6) = 4
      GO TO 33
   31 QC = QC2(JV)
      QT = QT2(JV)
      IJM = 4
      GAMMA = GAMMAC
      HEAT = HEATM(JV)
      NORMAL = 0
      ISKIP(6) = 3
      GO TO 33
   32 QC = QC1(JV)
      QT = QT1(JV)
      IJM = 1
      GAMMA = GAMMAI
      HEAT = HEATN(JV)
```

```
      NORMAL = 1
      ISKIP(6) = 2
   33 CONTINUE
      A = AA(JV)
      B = BB(JV)
      N = 3*JV-2
      M = N+2
      DO 35 I=1,3
      J = IJM+I-1
   35 LNTL(I) = LNCH(J)
      J = 3
      DO 36 I=N,M
      J = J+1
   36 LNTL(J) = TYPES(I)
      LNTL(7) = LNCH(10)
      LNTL(8) = LNCH(11)
      IF (HM .GT. 0.0) GO TO 40
      HM = Z(NZS)
   40 CONTINUE
      PRINT 2005, (LNTL(J),J=1,8)
      CALL CONSI(0,NO,YES,IPOL,ITP,IFEET,KNOTS)
C     CONVERT FEET TO METERS IF IFEET = F
      IF (IFEET .NE. ITP(1)) GO TO 76
      DO 75 K=1,NZS
   75 Z(K) = Z(K)*.3048
      HM = HM*.3048
      ZRK = ZRK*.3048
   76 CONTINUE
C     CONVERT KNOTS TO METERS/SEC IF KNOTS = K
      IF (KNOTS .NE. ITP(2)) GO TO 78
      DO 77 K=1,NZS
   77 WS(K) = WS(K)*.514791
   78 CONTINUE
   80 DO 81 K=1,NZS
      IF (Z(K)-1.0 .LT. HM.AND.HM .LT. Z(K)+1.0) GO TO 82
   81 CONTINUE
      GO TO 400
   82 KS = K
C     CONVERT TEMPERATURE FROM DEGREES CELSIUS TO ABSOLUTE
      DO 90 K=1,NZS
   90 T(K) = T(K)+273.16
```

```
C     CALCULATE VIRTUAL POTENTIAL TEMPERATURE                           VHC28700
      DO 100 K=1,NZS                                                    VHC28800
      XT = 1000.0/T(K)                                                  VHC28900
      XT = 8.42926604-XT*(1.82717843+.071208271*XT)                     VHC29000
      XT = RH(K)*.01*10.0**XT                                           VHC29010
      XT = 0.622*XT/(P(K)-XT)                                           VHC29100
      XT = T(K)*(1.0+1.61*XT)/(1.0+XT)                                  VHC29200
  100 TV(K) = CPHI(XT,P(K))                                             VHC29300
C     CALCULATE PLUME RISE                                              VHC29400
      IF (NORMAL .EQ. 0) GO TO 120                                      VHC29500
      CALL PLUME1                                                       VHC29600
      IF (IFLG .GT. 0) GO TO 410                                        VHC29700
      IF (JV .NE. 3) GO TO 130                                          VHC29800
      ZMSV = ZM                                                         VHC29900
      GAMMA = GAMMAC                                                    VHC30000
  120 CONTINUE                                                          VHC30100
      CALL PLUME2                                                       VHC30200
      IF (IFLG .GT. 0) GO TO 410                                        VHC30300
      IF (JV .NE. 3) GO TO 130                                          VHC30400
      IF (NORMAL .EQ. 0) GO TO 130                                      VHC30500
      GAMMA = .5*(GAMMAI+GAMMAC)                                        VHC30600
      ZM = .5*(ZM+ZMSV)                                                 VHC30700
      DO 121 I=2,NZS                                                    VHC30800
      IF (ZM .LT. Z(I)) GO TO 122                                       VHC30900
  121 CONTINUE                                                          VHC31000
  122 CALL LEAST(Z,TV,DPDZ,I,0,0.0,0.0)                                 VHC31100
      IF (DPDZ .LT. 3.322E-4) DPDZ = 3.322E-4                           VHC31200
  130 CONTINUE                                                          VHC31300
      IF (ISW(7) .NE. 0) CALL DELTXY                                    VHC31400
C     CALCULATE TURBULENCE PARAMETERS                                   VHC31500
      CALL TURB                                                         VHC31600
C     CALCULATE SOURCE DISTRIBUTION FOR MODEL 4                         VHC31700
      CALL DIST4                                                        VHC31800
C     CALCULATE SOURCE DIMENSIONS FOR MODEL 4                           VHC31900
      IFLG = 1                                                          VHC32000
      CALL DIM34                                                        VHC32100
      JBOT = 1                                                          VHC32200
      NNZ = NZS-1                                                       VHC32300
      JTOP = KS-1                                                       VHC32400
      DO 132 I=JBOT,JTOP                                                VHC32500
  132 IZMOD(I) = 4                                                      VHC32600
```

```
      IF (JTOP .GE. NNZ) GO TO 134                                      VHC32700
      J = JTOP+1                                                        VHC32800
      DO 133 I=J,NNZ                                                    VHC32900
  133 IZMOD(I) = 0                                                      VHC33000
  134 CONTINUE                                                          VHC33100
      XX = 1.0E3*22.4*1013.2*T(1)/(273.16*P(1))                         VHC33200
      I1 = 4                                                            VHC33300
C     OUTPUT NAMELIST NAM2 FOR HCL, CO, CO2, AL203 MODEL 4              VHC33400
      DO 200 I=1,4                                                      VHC33500
      ISKIP(5) = I                                                      VHC33600
      NT1 = 61                                                          VHC33700
      IF (I .EQ. 1) NT1 = 62                                            VHC33800
      IF (I .EQ. 3) NT1 = 69                                            VHC33900
      DO 135 J=1,10                                                     VHC34000
      DI(J) = DDI(J,I)                                                  VHC34100
      TI(J) = TII(J,I)                                                  VHC34200
  135 CI(J) = CCI(J,I)                                                  VHC34300
C     CALCULATE CONVERSTION FACTOR TO PPM FOR HCL, CO, CO2 AND TO       VHC34400
C     MILLIGRAMS PER CUBIC METER FOR AL203 AND ADJ FOR PERCENT OF MAT.  VHC34500
      IF (I .EQ. 4) GO TO 140                                           VHC34600
      QK = (XX/WTMOL(I))*FRQ(I)                                         VHC34700
      GO TO 150                                                         VHC34800
  140 QK = 1.0E3*FRQ(I)                                                 VHC34900
C     CONVERT Q TO PROPER UNITS AND PERCENTAGE OF POLLUTANT             VHC35000
  150 DO 160 K=1,NNZ                                                    VHC35100
  160 QF(K,I) = QK*Q(K)                                                 VHC35200
      IF (ISW(1) .EQ. 0) GO TO 200                                      VHC35300
      IF (ISW(3) .EQ. 0.AND.I .EQ. 1) GO TO 200                         VHC35400
      IF (ISW(4) . EQ. 0 .AND.I .EQ. 2) GO TO 200                       VHC35500
      IF (ISW(5) .EQ. 0.AND.I .EQ. 4) GO TO 200                         VHC35600
      IF (ISW(6) .EQ. 0.AND.I .EQ. 3) GO TO 200                         VHC35700
      PRINT 2005,NAMCAS,(LNTL(J),J=1,8)                                 VHC35800
      PRINT 2007                                                        VHC35900
      K = 1                                                             VHC36000
      IF (I .EQ. 4) K = 3                                               VHC36100
      IF (ISW(2) .GT. 0) GO TO 170                                      VHC36200
      IF (IHM .NE. IBNK) GO TO 170                                      VHC36300
      IF (I .EQ. IPNPS) NPS = 1                                         VHC36400
  170 CONTINUE                                                          VHC36500
      PRINT 2001, (DATE(J),J=1,6),(TYPES(J),J=N,M)                      VHC36600
      PUNCH 2001, (DATE(J),J=1,6),(TYPES(J),J=N,M)                      VHC36700
```

```
  180 CALL OUTPT(KS,I,4)                                                    VHC36800
  200 CONTINUE                                                              VHC36900
      IF (ISW(1) .EQ. 0) GO TO 205                                          VHC37000
      PRINT 2005, NAMCAS,(LNTL(J),J=1,8)                                    VHC37100
      CALL CONST(4,NO,YES,IPOL,ITP,IFEET,KNOTS)                             VHC37200
  205 CONTINUE                                                              VHC37300
C     OUTPUT NAMELIST NAM2 FOR HCL, CO, CO2, AL2O3 MODEL 3                  VHC37400
C     CALCULATE SOURCE DISTRIBUTION FOR MODEL 3                             VHC37500
      CALL DIST3                                                            VHC37600
C     CALCULATE SOURCE DIMENSIONS FOR MODEL 3                               VHC37700
      IFLG = 0                                                              VHC37800
      CALL DIM34                                                            VHC37900
      Z(2) = HM                                                             VHC38000
      SIGAP(2) = SIGAP(KS)                                                  VHC38100
      SIGEP(2) = SIGEP(KS)                                                  VHC38200
      WD(2) = WD(KS)                                                        VHC38300
      UBAR(2) = UBAR(KS)                                                    VHC38400
      IZMOD(1) = 3                                                          VHC38500
      DX(1) = DX(ILXY)                                                      VHC38600
      DY(1) = DY(ILXY)                                                      VHC38700
      IF (SIGZO(1) .LE. 0.0) GO TO 420                                      VHC38800
      NNZ = 1                                                               VHC38900
      NZS = 2                                                               VHC39000
      KS = 2                                                                VHC39100
      II = 3                                                                VHC39200
      DO 260 I=1,4                                                          VHC39300
      ISKIP(5) = I                                                          VHC39400
      NTI = 61                                                              VHC39500
      IF (I .EQ. 1) NTI = 62                                                VHC39600
      IF (I .EQ. 3) NTI = 69                                                VHC39700
      DO 210 J=1,10                                                         VHC39800
      DI(J) = DDI(J,I)                                                      VHC39900
      TI(J) = TII(J,I)                                                      VHC40000
  210 CI(J) = CCI(J,I)                                                      VHC40100
C     CALCULATE CONVERSION FACTOR TO PPM FOR HCL, CO, CO2 AND TO            VHC40200
C     MILLIGRAMS PER CUBIC METER FOR AL2O3 AND ADJ FOR PERCENT OF MAT.      VHC40300
      IF (I .EQ. 4) GO TO 220                                               VHC40400
      QF(1,I) = Q(1)*(XX/WTMOL(I))*FRQ(I)                                   VHC40500
      GO TO 230                                                             VHC40600
  220 QF(1,1) = Q(1)*1.0E3*FRQ(I)                                           VHC40700
  230 CONTINUE                                                              VHC40800
```

```
      IF (ISW(2) .EQ. 0) GO TO 260                                      VHC40900
      IF (ISW(3) .EQ. 0.AND.I .EQ. 1) GO TO 260                         VHC41000
      IF (ISW(4) .EQ. 0.AND.I .EQ. 2) GO TO 260                         VHC41100
      IF (ISW(5) .EQ. 0.AND.I .EQ. 4) GO TO 260                         VHC41200
      IF (ISW(6) .EQ. 0.AND.I .EQ. 3) GO TO 260                         VHC41300
      PRINT 2005,NAMCAS,(LNTL(J),J=1,8)                                 VHC41400
      PRINT 2007                                                        VHC41500
      K = 1                                                             VHC41600
      IF (I .EQ. 4) K = 3                                               VHC41700
      IF (IHM .NE. IBNK) GO TO 240                                      VHC41800
      IF (I .EQ. IPNPS) NPS = 1                                         VHC41900
  240 CONTINUE                                                          VHC42000
      PRINT 2001, (DATE(J),J=1,6),(TYPES(J),J=N,M)                      VHC42100
      PUNCH 2001, (DATE(J),J=1,6),(TYPES(J),J=N,M)                      VHC42200
  250 CALL OUTP1(KS,I,3)                                                VHC42300
  260 CONTINUE                                                          VHC42400
      IF (ISW(2) .EQ. 0) GO TO 500                                      VHC42500
      PRINT 2005, NAMCAS,(LNTL(J),J=1,8)                                VHC42600
      CALL CONST(3,NO,YES,IPOL,ITP,IFEET,KNOTS)                         VHC42700
      GO TO 500                                                         VHC42800
  400 PRINT 2003                                                        VHC42900
      GO TO 500                                                         VHC43000
  410 II = 2                                                            VHC43100
      IF (IFEET .NE. ITP(1)) GO TO 411                                  VHC43200
      II = 1                                                            VHC43300
      ZM = ZM/.3048                                                     VHC43400
  411 PRINT 2004, ZM,JP(II)                                             VHC43500
      GO TO 500                                                         VHC43600
  420 PRINT 2006                                                        VHC43700
  500 IF (IHM .NE. IBNK) GO TO 5                                        VHC43800
      STOP                                                              VHC43900
 1000 FORMAT (1X,2A3,2A1,6A6,2F10.0,1X,7I1)                             VHC44000
 1001 FORMAT (6F10.4,19X,A1)                                            VHC44100
 1002 FORMAT (12A6)                                                     VHC44200
 2001 FORMAT (6H $NAM2/11H TESTNO=60H,9A6,7H      ,)                    VHC44300
 2003 FORMAT (84H0 **ERROR** HM MUST BE EQUAL TO ONE OF THE LAYER BOUNDAVHC44400
     1RIES Z AND IN THE SAME UNITS./)                                   VHC44500
 2004 FORMAT  (70H0 **ERROR** NOT ENOUGH LAYERS, TOP OF LAST LAYER MUST VHC44600
     1BE GREATER THAN ,1PE12.5,1X,A6,19H, INPUT MORE LAYERS/)           VHC44700
 2005 FORMAT (1H1,21X,8H*-*-*-* ,12A6,8H *-*-*-*/29X,8H*-*-*-* ,3A6,1X,5VHC44800
     1A6,8H *-*-*-*/)                                                   VHC44900
```

```
      2006 FORMAT (58H ***ERROR***, COLUMNS 1-3 ARE INCORRECTLY PUNCHED ON CARVHC45000
          1D 1)                                                                VHC45100
      2007 FORMAT (28X,75H*-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILVHC45200
          1AYER MODEL VERSION 5 *-*//)                                         VHC45300
      2008 FORMAT (102H1*-* CLOUD RISE IS WELL ABOVE HM. MODEL 3 PARAMETERS AVHC45400
          1RE NOT CALCULATED. USE MODEL 4 FOR THIS CASE *-*)                   VHC45500
      2009 FORMAT (54H *WARNING* VEHICLE TYPE NOT SPECIFIED, TITAN IIIC USED)VHC45600
           END                                                                 VHC45700
```

```
      SUBROUTINE CONV(Q,I,J,K)                                          CNV00100
      DIMENSION ICHAR(12)                                               CNV00200
      DATA ICHAR/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H+,1H-/       CNV00300
      R = 0.0                                                           CNV00400
      IP = 0                                                            CNV00500
      K = ICHAR(11)                                                     CNV00600
      IF (Q) 100,55,10                                                  CNV00700
   10 IF (Q-1.0) 20,30,40                                               CNV00800
   20 IP = 0                                                            CNV00900
      R = Q                                                             CNV01000
   21 IF (R .GT. 1.0) GO TO 50                                          CNV01100
      IP = IP-1                                                         CNV01200
      R = R*10.0                                                        CNV01300
      GO TO 21                                                          CNV01400
   30 R = Q                                                             CNV01500
      IP = 0                                                            CNV01600
      GO TO 55                                                          CNV01700
   40 IP = 0                                                            CNV01800
      R = Q                                                             CNV01900
   41 IF (R .LT. 10.0) GO TO 50                                         CNV02000
      IP = IP+1                                                         CNV02100
      R = R*0.1                                                         CNV02200
      GO TO 41                                                          CNV02300
   50 CONTINUE                                                          CNV02400
      Q = R                                                             CNV02500
      K = ICHAR(11)                                                     CNV02600
      IF (IP .LT. 0) K = ICHAR(12)                                      CNV02700
   55 I = IABS(IP/10)                                                   CNV02800
      J = IABS(IP)-I*10                                                 CNV02900
   60 DO 70 L=1,10                                                      CNV03000
      IF (I .NE. L-1) GO TO 70                                          CNV03100
      I = ICHAR(L)                                                      CNV03200
      GO TO 71                                                          CNV03300
   70 CONTINUE                                                          CNV03400
      GO TO 110                                                         CNV03500
   71 CONTINUE                                                          CNV03600
      DO 80 L=1,10                                                      CNV03700
      IF (J .NE. L-1) GO TO 80                                          CNV03800
      J = ICHAR(L)                                                      CNV03900
      GO TO 81                                                          CNV04000
```

```
   80 CONTINUE                                                        CNV04100
      GO TO 110                                                       CNV04200
   81 CONTINUE                                                        CNV04300
   90 RETURN                                                          CNV04400
  100 PRINT 2000, Q                                                   CNV04500
      GO TO 90                                                        CNV04600
  110 PRINT 2001, Q                                                   CNV04700
      GO TO 90                                                        CNV04800
 2000 FORMAT (39H *-*ERROR*-* SOURCE STRENGTH NEGATIVE =,E15.8)       CNV04900
 2001 FORMAT (47H *-*ERROR*-* POWER OF 10 ON Q TOO MANY DIGITS =,E15.8) CNV05000
      END                                                             CNV05100
```

```
      SUBROUTINE TURB                                                   TRB00100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(TRB00200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             TRB00300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          TRB00400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           TRB00500
      PHI1 = G*DPDZ/T(1)                                                TRB00600
      TAUK = PI/SQRT(PHI1)                                              TRB00700
      IF (TAUK .GT. 600.0) TAUK = 600.0                                 TRB00800
      K = 0                                                             TRB00900
   10 K = K+1                                                           TRB01000
      IF (K .GT. NZS) GO TO 40                                          TRB01100
      IF (Z(K) .GT. HM) GO TO 35                                        TRB01200
      SIGAP(K) = SIGAP(1)                                               TRB01300
      SIGEP(K) = SIGAP(1)                                               TRB01400
      GO TO 10                                                          TRB01500
   35 SIGAP(K) = 0.1                                                    TRB01600
      SIGEP(K) = 0.1                                                    TRB01700
      GO TO 10                                                          TRB01800
   40 CONTINUE                                                          TRB01900
      RETURN                                                            TRB02000
      END                                                               TRB02100
```

```
      SUBROUTINE DIM34                                                  D3400100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(D3400200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             D3400300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          D3400400
      COMMON /SIG/ SIGXO(20),SIGYO(20),SIGZO(20),H                      D3400500
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           D3400600
      IF(IFLG .EQ. 1) GO TO 30                                          D3400700
C     SOURCE DIMENSIONS FOR MODEL 3                                     D3400800
      IF (ZM .LT. HM-GAMMA*ZM) GO TO 10                                 D3400900
      SIGZO(1) = (HM-ZM+GAMMA*ZM)*.2325581                              D3401000
      H = (HM+ZM-GAMMA*ZM)*0.5                                          D3401100
      GO TO 20                                                          D3401200
   10 SIGZO(1) = GAMMA*ZM*.465116279                                    D3401300
      H = ZM                                                            D3401400
   20 SIGXO(1) = GAMMA*ZM*.465116279                                    D3401500
      SIGYO(1) = SIGXO(1)                                               D3401600
      IF (SIGZO(1) .GT. 0.0) GO TO 50                                   D3401700
      H = 0.5*(HM-Z(1))                                                 D3401800
      SIGZO(1) = GAMMA*H*.465116279                                     D3401900
      GO TO 50                                                          D3402000
C     SOURCE DIMENSIONS FOR MODEL 4                                     D3402100
   30 DO 40 K=2,NZS                                                     D3402200
      ZK = Z(K-1)                                                       D3402300
      IF (K .EQ. 2) ZK = 0.0                                            D3402400
      ZP = 0.5*(Z(K)-ZK)+ZK                                             D3402500
      IF (ZP .GT. ZM) GO TO 35                                          D3402600
      SXO = ZM                                                          D3402700
      GO TO 36                                                          D3402800
   35 SXO = (2.0*ZM-ZP)                                                 D3402900
   36 SXO = SXO*GAMMA*.465116279                                        D3403000
      IF (SXO .LT. 0.0) SXO = 0.0                                       D3403100
      IF (NORMAL .EQ. 0) GO TO 38                                       D3403200
      IF (ZP .LE. ZM) GO TO 38                                          D3403300
      IF (SXO .LT. 93.0) SXO = 93.0                                     D3403400
   38 SIGXO(K-1) = SXO                                                  D3403500
      SIGYO(K-1) = SXO                                                  D3403600
      SIGZO(K-1) = 0.0                                                  D3403700
   40 CONTINUE                                                          D3403800
      H = ZM                                                            D3403900
   50 RETURN                                                            D3404000
      END                                                               D3404100
```

```
      FUNCTION CPHI(A,B)                                              PHI00100
      CPHI = A*(1000.0/B)**0.288                                      PHI00200
      RETURN                                                          PHI00300
      END                                                             PHI00400
```

```
      FUNCTION TPZ(A,B,C,D,E)                                         TPZ00100
      TPZ = (A-B)*(C-D)/(A-E)                                         TPZ00200
      RETURN                                                          TPZ00300
      END                                                             TPZ00400
```

```
      SUBROUTINE DIST4                                                  DS400100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(DS400200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             DS400300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          DS400400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           DS400500
      IF (NORMAL .EQ. 0) GO TO 5                                        DS400600
      QQ = QC*A*ZM**B                                                   DS400700
      GO TO 6                                                           DS400800
    5 QQ = QT                                                           DS400900
    6 SQ2I = 1.0/(1.41421356*GAMMA*ZM*.465116279)                       DS401000
      K = 1                                                             DS401100
      PHI = 0.0                                                         DS401200
   10 K = K+1                                                           DS401300
      IFLG = 0                                                          DS401400
      ZP = (Z(K)-ZM)*SQ2I                                               DS401500
      IF (ZP) 20,15,30                                                  DS401600
   15 CONTINUE                                                          DS401700
      PZ = 0.5                                                          DS401800
      GO TO 60                                                          DS401900
   20 CONTINUE                                                          DS402000
      ZP = -ZP                                                          DS402100
      IFLG = 1                                                          DS402200
   30 A1 = 0.5/(ZP*ZP)                                                  DS402300
      TZ = 1.0                                                          DS402400
      IF (ZP .GT. 6.99) GO TO 50                                        DS402500
      B1 = ZP*1.12837917/EXP(ZP*ZP)                                     DS402600
      TZ = B1                                                           DS402700
   40 A1 = A1+1.0/(ZP*ZP)                                               DS402800
      PHI1 = B1/A1                                                      DS402900
      TZ = TZ+PHI1                                                      DS403000
      IF (PHI1 .LE. 1.0E-10) GO TO 50                                   DS403100
      B1 = PHI1                                                         DS403200
      GO TO 40                                                          DS403300
   50 PZ = 0.5*(1.0+TZ)                                                 DS403400
      IF (IFLG .EQ. 1) PZ = 1.0-PZ                                      DS403500
   60 Q(K-1) = (PZ-PHI)*QQ                                              DS403600
      IF (Q(K-1) .LT. 0.0) Q(K-1) = 0.0                                 DS403700
      IF (Q(K-1) .LT. 1.0E-20) QQ = 0.0                                 DS403800
      PHI = PZ                                                          DS403900
      IF (K .LT. NZS) GO TO 10                                          DS404000
```

```
      IF (NORMAL .EQ. 0) GO TO 90                                       DS404100
      K = 2                                                             DS404200
      ZP = ZM                                                           DS404300
   70 IF (Z(K) .GE. ZM) GO TO 80                                        DS404400
      K = K+1                                                           DS404500
      IF (K .LE. NZS) GO TO 70                                          DS404600
      GO TO 90                                                          DS404700
   80 IF (K .GT. NZS) GO TO 90                                          DS404800
      Q(K-1) = QC*A*(Z(K)**B-ZP**B)+Q(K-1)                              DS404900
      ZP = Z(K)                                                         DS405000
      K = K+1                                                           DS405100
      GO TO 80                                                          DS405200
   90 CONTINUE                                                          DS405300
      RETURN                                                            DS405400
      END                                                               DS405500
```

```
      SUBROUTINE DIST3                                                  DS300100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(DS300200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             DS300300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          DS300400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           DS300500
      IF (NORMAL .EQ. 0) GO TO 2                                        DS300600
      QQ = QC*A*ZM**B                                                   DS300700
      GO TO 3                                                           DS300800
    2 QQ = QT                                                           DS300900
    3 CONTINUE                                                          DS301000
      IFLG = 0                                                          DS301100
      ZP = (HM-ZM)/(1.41421356*GAMMA*ZM*.465116279)                     DS301200
      PZ = 0.5                                                          DS301300
      IF (ZP) 5,40,10                                                   DS301400
    5 CONTINUE                                                          DS301500
      ZP = -ZP                                                          DS301600
      IFLG = 1                                                          DS301700
   10 A1 = 0.5/(ZP*ZP)                                                  DS301800
      TZ = 1.0                                                          DS301900
      IF (ZP .GT. 6.99) GO TO 30                                        DS302000
      B1 = ZP*1.12837917/EXP(ZP*ZP)                                     DS302100
      TZ = B1                                                           DS302200
   20 A1 = A1+1.0/(ZP*ZP)                                               DS302300
      PHI1 = B1/A1                                                      DS302400
      TZ = TZ+PHI1                                                      DS302500
      IF (PHI1 .LE. 1.0E-10) GO TO 30                                   DS302600
      B1 = PHI1                                                         DS302700
      GO TO 20                                                          DS302800
   30 PZ = 0.5*(1.0+TZ)                                                 DS302900
      IF (IFLG .EQ. 1) PZ = 1.0-PZ                                      DS303000
   40 Q(1) = PZ*QQ                                                      DS303100
      RETURN                                                            DS303200
      END                                                               DS303300
```

```
      SUBROUTINE OUTPT(NZ,I,IK)                                         OPT00100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(OPT00200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             OPT00300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          OPT00400
      COMMON /SIG/ SIGXO(20),SIGYO(20),SIGZO(20),H                      OPT00500
      COMMON /OUT/ QF(20,4),WD(21),ISKIP(10),NDI,NCI,NTI,ZRK,JBOT,JTOP, OPT00600
     1IZMOD(21),CI(10),DI(10),TI(10),NPS                                OPT00700
      COMMON /DISPL/ DXX,DX(21),DYY,DY(21),ILXY                         OPT00800
      DIMENSION QL(20),IQ(20),JQ(20),KQ(20)                             OPT00900
      DIMENSION JE(15)                                                  OPT01000
      DATA JE/90H     Z     Q UBARK SIGAK SIGEK SIGXO SIGYO SIGZOTHETAK OPT01100
     1   DI    CI  DELX  DELY    TI TEMPK/                              OPT01200
      NZM = NZ                                                          OPT01300
      IF (NZ .GT. 16) NZ = 16                                           OPT01400
      NNZ = NZ-1                                                        OPT01500
      NDII = NDI/10                                                     OPT01600
      NCII = NCI/10                                                     OPT01700
      NTII = NTI/10                                                     OPT01800
      ZF = 0.0                                                          OPT01900
      TIMAV = 600.0                                                     OPT02000
      IF (I .EQ. 2) TIMAV = 360.0                                       OPT02100
      PRINT 1999, NAMCAS                                                OPT02110
 1999 FORMAT (1X,10HNAMCAS=68H,11A6,A2,1H,)                             OPT02220
      PRINT 2000, ISKIP,NPS,NZ,NDI,NCI,NTI,ZRK,TAUK,(IZMOD(K),K=1,NNZ)  OPT02200
      PRINT 2001, JE(1),ZF,(Z(K),K=2,NZ)                                OPT02300
      PRINT 2002,     JE(2),(QF(K,I),K=1,NNZ)                           OPT02400
      PRINT 2001,     JE(3),(UBAR(K),K=1,NZ)                            OPT02500
      PRINT 2001,     JE(4),(SIGAP(K),K=1,NZ)                           OPT02600
      PRINT 2001,     JE(5),(SIGEP(K),K=1,NZ)                           OPT02700
      PRINT 2001,     JE(6),(SIGXO(K),K=1,NNZ)                          OPT02800
      PRINT 2001,     JE(7),(SIGYO(K),K=1,NNZ)                          OPT02900
      PRINT 2001,     JE(8),(SIGZO(K),K=1,NNZ)                          OPT03000
      PRINT 2001,     JE(9),(WD(K),K=1,NZ)                              OPT03100
      PRINT 2006, TIMAV                                                 OPT03200
      PRINT 2001,     JE(10),(DI(K),K=1,NDII)                           OPT03300
      PRINT 2001,     JE(11),(CI(K),K=1,NCII)                           OPT03400
      PRINT 2001, JE(14),(TI(K),K=1,NTII)                               OPT03500
      PRINT 2001, JE(12),(DX(K),K=1,NNZ)                                OPT03600
      PRINT 2001, JE(13),(DY(K),K=1,NNZ)                                OPT03700
      PRINT 2001, JE(15),(TV(K),K=1,NZ)                                 OPT03800
```

```
      PRINT 2005, H                                                     OPT03900
      PRINT 2003                                                        OPT04000
 2000 FORMAT (7H ISKIP=,10(I1,1H,),4HNPS=,I2,5H,NZS=,I2,5H,NDI=,I2,5H,NCOPT04100
     1I=,I2,5H,NTI=,I2,5H,ZRK=,F5.1,1H,/6H TAUK=,F8.3,7H,IZMOD=,15(I1,1HOPT04200
     2,))                                                               OPT04300
 2001 FORMAT (1X,A6,1H=,7(F9.3,1H,)/(1X,7(F9.3,1H,)))                   OPT04400
 2002 FORMAT (1X,A6,1H=,1P4(E14.7,1H,)/1X,5(E14.7,1H,)/(1X,5(E14.7,1H,))OPT04500
     1)                                                                 OPT04600
 2003 FORMAT (5H $END)                                                  OPT04700
 2004 FORMAT (1X,A6,1H=,4(F11.8,1HE,3A1,1H,)/1X,4(F11.8,1HE,3A1,1H,)/   OPT04800
     11X,4(F11.8,1HE,3A1,1H,))                                          OPT04900
 2005 FORMAT (3H H=,F9.3,1H,)                                           OPT05000
 2006 FORMAT (7H TIMAV=,F6.1,1H,)                                       OPT05100
      PUNCH 1999, NAMCAS                                                OPT05110
      PUNCH 2000, ISKIP,NPS,NZ,NDI,NCI,NTI,ZRK,TAUK,(IZMOD(K),K=1,NNZ)  OPT05200
      PUNCH 2001, JE(1),ZF,(Z(K),K=2,NZ)                                OPT05300
      DO 20 K=1,NNZ                                                     OPT05400
      QL(K) = QF(K,I)                                                   OPT05500
   20 CALL CONV(QL(K),IQ(K),JQ(K),KQ(K))                                OPT05600
      PUNCH 2004, JE(2),(QL(K),KQ(K),IQ(K),JQ(K),K=1,NNZ)               OPT05700
      PUNCH 2001, JE(3),(UBAR(K),K=1,NZ)                                OPT05800
      PUNCH 2001, JE(4),(SIGAP(K),K=1,NZ)                               OPT05900
      PUNCH 2001, JE(5),(SIGEP(K),K=1,NZ)                               OPT06000
      PUNCH 2001, JE(6),(SIGXO(K),K=1,NNZ)                              OPT06100
      PUNCH 2001, JE(7),(SIGYO(K),K=1,NNZ)                              OPT06200
      PUNCH 2001, JE(8),(SIGZO(K),K=1,NNZ)                              OPT06300
      PUNCH 2001, JE(9),(WD(K),K=1,NZ)                                  OPT06400
      PUNCH 2006, TIMAV                                                 OPT06500
      PUNCH 2001, JE(10),(DI(K),K=1,NDII)                               OPT06600
      PUNCH 2001, JE(11),(CI(K),K=1,NCII)                               OPT06700
      PUNCH 2001, JE(14),(TI(K),K=1,NTII)                               OPT06800
      PUNCH 2001, JE(12),(DX(K),K=1,NNZ)                                OPT06900
      PUNCH 2001, JE(13),(DY(K),K=1,NNZ)                                OPT07000
      PUNCH 2001, JE(15),(TV(K),K=1,NZ)                                 OPT07100
      PUNCH 2005, H                                                     OPT07200
      PUNCH 2007, TAUK                                                  OPT07300
 2007 FORMAT (1X,6HTAUOK=,F8.3,1H,)                                     OPT07400
      PUNCH 2003                                                        OPT07500
      NZ = NZM                                                          OPT07600
      RETURN                                                            OPT07700
      END                                                               OPT07800
```

```
      SUBROUTINE CONST(JFLG,NO,YES,IPOL,ITP,IFEET,KNOTS)                CST00100
      COMMON /PLUME/QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(2CST00200
     11),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL              CST00300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          CST00400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           CST00500
      COMMON /SIG/ SIGXO(20),SIGYO(20),SIGZO(20),H                      CST00600
      COMMON /F/ DATE(6),FRQ(4),WTMOL(3)                                CST00700
      COMMON /OUT/ QF(20,4),WD(21),ISKIP(10),NDI,NCI,NTI,ZRK,JBOT,JTOP, CST00800
     11ZMOD(21),CI(10),DI(10),TI(10),NPS                                CST00900
      COMMON /DISPL/ DXX,DX(21),DYY,DY(21),ILXY                         CST01000
      DIMENSION IFQ(4)                                                  CST01100
      DATA IFQ/24H FEET METERSKNOTS MET/S /                             CST01200
      DIMENSION IPOL(4)                                                 CST01300
      DIMENSION ITP(2)                                                  CST01400
      INTEGER YES                                                       CST01500
      IF (JFLG .GT. 0) GO TO 20                                         CST01600
      PRINT 2000, QC,QT,A,B,HEAT,GAMMA,CP                               CST01700
 2000 FORMAT (1H0,41X,     47H*-* INITIALIZED DATA USED FOR ABOVE VEHICLCST01800
     1E *-*//20X,69HQC - RATE OF OUTPUT OF EXHAUST MATERIAL FROM VEHICLECST01900
     2 IN GRAMS/SEC IS ,1PE15.8/                                        CST02000
     320X,58HQT - TOTAL AMOUNT OF VEHICLE EXHAUST MATERIAL IN GRAMS IS ,CST02100
     4E15.8/                                                            CST02200
     520X,64HA AND B - VEHICLE RISE PARAMETERS IN THE EQUATION TR=A*Z**BCST02300
     6 ARE ,0PF8.6,5H AND ,F8.6/                                        CST02400
     720X,45HHEAT - TOTAL HEAT OUTPUT IN CALORIES/GRAM IS ,F10.4/       CST02500
     820X,33HGAMMA - ENTRAINMENT PARAMETER IS ,F7.4/                    CST02600
     920X,29HCP - SPECIFIC HEAT OF AIR IS ,F5.3)                        CST02700
      PRINT 2001, IPOL,FRQ,WTMOL                                        CST02800
 2001 FORMAT (20X,23HPOLUTANT MATERIALS ARE ,4(A6,1H,)/                 CST02900
     120X,50HFRACTIONAL DISTRIBUTION OF THE ABOVE MATERIALS IS ,4(F5.3, CST03000
     21H,)/                                                             CST03100
     320X,43HMOLECULAR WEIGHT OF THE ABOVE MATERIALS IS ,3(F7.3,1H,)  ) CST03200
      J1 = NO                                                           CST03300
      IF (NORMAL .EQ. 1) J1 = YES                                       CST03400
      J2 = NO                                                           CST03500
      IF (IFEET .EQ. ITP(1)) J2 = YES                                   CST03600
      J3 = NO                                                           CST03700
      IF (KNOTS .EQ. ITP(2)) J3 = YES                                   CST03800
      N1 = IFQ(2)                                                       CST03900
      IF (J2 .EQ. YES) N1 = IFQ(1)                                      CST04000
```

```
      N2 = IFQ(4)                                                       CST04100
      IF (J3 .EQ. YES) N2 = IFQ(3)                                      CST04200
      PRINT 2002, DATE,J1,J2,J3,SIGAR,RHO,HM,NZS                        CST04300
 2002 FORMAT (1H0,52X,26H*-X PROGRAM INPUT DATA *-*//25X,11HDATA CARD 1/CST04400
     129X,8HTITLE - ,6A6/29X,28HNORMAL - IS LAUNCH NORMAL ? ,A3/        CST04500
     229X,54HIFEET - ARE LAYER BOUNDARY HEIGHTS Z, AND HM IN FEET? ,A3/ CST04600
     329X,39HKNOTS - IS THE WIND SPEED WS IN KNOTS? ,A3/                CST04700
     429X,67HSIGAR - STANDARD DEVIATION OF THE AZIMUTH WIND ANGLE IN DEGCST04800
     5REES IS ,F7.3/                                                    CST04900
     629X,42HRHO - AIR DENSITY IN GRAMS/CUBIC METER IS ,F9.3/           CST05000
     729X,39HHM - HEIGHT OF SURFACE MIXING LAYER IS ,F9.3/              CST05100
     825X,20HDATA CARD 2 THROUGH ,I2/33X,67HLAYER BOUNDARY   WIND DIRECTCST05200
     9ION  WIND SPEED  TEMPERATURE   PRESSURE)                          CST05300
      PRINT 2003, N1,N2                                                 CST05400
 2003 FORMAT (39X,1HZ,1X,1H(,A6,1H),4X,9HWD  (DEG),5X,2HWS,1X,1H(,A5,1H)CST05500
     1,2X,9HT (DEG C),6X,6HP (MB),3X,12HRH (PERCENT))                   CST05600
      DO 10 I=1,NZS                                                     CST05700
   10 PRINT 2004, Z(I),WD(I),UBAR(I),T(I),P(I),RH(I)                    CST05800
 2004 FORMAT (34X,F9.3,9X,F9.4,4X,F9.4,5X,F9.3,3X,F9.3,5X,F7.3)         CST05900
      GO TO 40                                                          CST06000
   20 CONTINUE                                                          CST06100
      NN2 = NZS-1                                                       CST06200
      ZZL = 0.0                                                         CST06300
      PRINT 2005, JFLG,H,ZM,TAUK,DPDZ,JBOT,JTOP,ZZL                     CST06400
 2005 FORMAT (1H0,39X,36H*-* CALCULATED PARAMETERS FOR MODEL ,I2,4H *-*/CST06500
     1/13X,39HH - ADJUSTED CLOUD HEIGHT IN METERS IS ,F9.3/             CST06600
     213X,36HZM - REAL CLOUD HEIGHT IN METERS IS ,F9.3/                 CST06700
     313X,45HTAUK - TIME TO CLOUD STABILIZATION IN SEC IS ,F9.3/        CST06800
     413X,73HDPDZ - VERTICAL GRADIENT OF AMBIENT POTENTIAL TEMP IN DEGRECST06900
     5ES K/METER IS ,F12.6/                                             CST07000
     613X,44HJBOT - BOTTOM LAYER FOR USE WITH MODEL 4 IS ,I2/           CST07100
     713X,41HJTOP - TOP LAYER FOR USE WITH MODEL 4 IS ,I2/              CST07200
     813X,58HZ - BOUNDARY HEIGHT AT THE BOTTOM OF LAYER 1 IN METERS IS ,CST07300
     9F8.3)                                                             CST07400
      PRINT 2006, ZRK,SIGAP(1),SIGEP(1),IPOL                            CST07500
 2006 FORMAT (13X,83HSIGAP - STANDARD DEVIATION OF THE WIND AZIMUTH ANGLCST07600
     1E AT THE MEASUREMENT HEIGHT ZRK=,F6.2,11H METERS IS ,F8.3/        CST07700
     213X,65HSIGEP - STANDARD DEVIATION OF THE WIND ELEVATION ANGLE AT ZCST07800
     3RK IS ,F6.3/19H LAYER PARAMETERS -/11H LAYER    Z,21X,21H- SOURCE CST07900
     4STRENGTH Q -,19X,59HSIGAP   SIGEP    SIGXO    SIGYO    SIGZO      CST08000
     5 DELX    DELY/13H  NO.  (LAYER,59X,59H(DEG)   (DEG)   (METER)   (MCST08100
```

```
     6ETER)    (METER)   (METER)   (DEG)/9X,4HTOP),4X,A6,9X,A6,8X,A6,8X,A6/CST08200
     7)                                                                  CST08300
      DO 30 K=1,NNZ                                                      CST08400
   30 PRINT 2007, K,Z(K+1),(QF(K,I),I=1,4),SIGAP(K+1),SIGEP(K+1),SIGXO(KCST08500
     1),SIGYO(K),SIGZO(K),DX(K),DY(K)                                    CST08600
 2007 FORMAT (1X,I3,1X,F9.3,1P4E14.7,0P2F8.4,3F10.4,F9.2,F7.2)           CST08700
   40 CONTINUE                                                           CST08800
      RETURN                                                             CST08900
      END                                                                CST09000
```

```
      SUBROUTINE PLUME1                                                  PL100100
      COMMON /PLUME/ QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(PL100200
     121),NZS,G(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             PL100300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          PL100400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS            PL100500
C     PLUME RISE FOR INSTANTANEOUS SOURCE                                PL100600
      A1 = 6.0*QC*A*HEAT/(RHO*CP*PI*GAMMA**3)                            PL100700
      B1 = 1.0/(4.0-B)                                                   PL100800
      K = 1                                                              PL100900
   10 K = K+1                                                            PL101000
   20 CALL LEAST(Z,TV,DPDZ,K,0,0.0,0.0)                                  PL101100
      IF (DPDZ .LT. 3.322E-4) DPDZ = 3.322E-4                            PL101200
      ZM = (A1/DPDZ)**B1                                                 PL101300
      IF (ZM .LE. Z(K)) GO TO 30                                         PL101400
      K = K+1                                                            PL101500
      IF (K .GT. NZS) GO TO 80                                           PL101600
      GO TO 20                                                           PL101700
   30 IF (Z(K)-ZM .LE. 10.0) GO TO 70                                    PL101800
      IF (DPDZ-3.322E-4) 35,70,35                                        PL101900
   35 CONTINUE                                                           PL102000
      ZP = Z(K)                                                          PL102100
   40 ZP = ZP-10.0                                                       PL102200
      IF (ZP .LT. Z(1)) GO TO 65                                         PL102300
      TVP = TV(K)-TPZ(Z(K),ZP,TV(K),TV(K-1),Z(K-1))                      PL102400
      CALL LEAST(Z,TV,DPDZ,K-1,1,ZP,TVP)                                 PL102500
      IF (DPDZ .GT. 3.322E-4) GO TO 60                                   PL102600
      DPDZ = 3.322E-4                                                    PL102700
   50 ZM = ZP                                                            PL102800
      GO TO 70                                                           PL102900
   60 ZM = (A1/DPDZ)**B1                                                 PL103000
      IF (ZM .GT. ZP) GO TO 50                                           PL103100
      IF (ZM .GT. ZP-10.0) GO TO 70                                      PL103200
      IF (ZP .GE. Z(K-1)) GO TO 40                                       PL103300
      ZM = Z(K-1)                                                        PL103400
C     RETURN ZM AND DPDZ FOR INSTANTANEOUS SOURCE                        PL103500
   70 IFLG = 0                                                           PL103600
      GO TO 90                                                           PL103700
C     CANNOT CALCULATE ZM AND DPDZ                                       PL103800
   80 IFLG = 1                                                           PL103900
      GO TO 90                                                           PL104000
```

```
   65 IFLG = 2                                                        PL104100
   90 RETURN                                                          PL104200
      END                                                             PL104300
```

```
      SUBROUTINE PLUME2                                                 PL200100
      COMMON /PLUME/ GC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR(PL200200
     121),NZS,G(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             PL200300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                          PL200400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS           PL200500
C     PLUME RISE FOR CONTINUOUS SOURCE                                  PL200600
      ZSUM = 0.0                                                        PL200700
      UBARS = 0.0                                                       PL200800
      A1 = 6.0*GC*HEAT/(RHO*CP*PI*GAMMA**2)                             PL200900
      B1 = .3333333                                                     PL201000
      K = 1                                                             PL201100
   10 K = K+1                                                           PL201200
   20 CALL LEAST(Z,TV,DPDZ,K,0,0.0,0.0)                                 PL201300
      IF (DPDZ .LT. 3.322E-4) DPDZ = 3.322E-4                           PL201400
      UBARS = UBARS+(Z(K)-Z(K-1))*(UBAR(K)+UBAR(K-1))*0.5               PL201500
      ZSUM = ZSUM+Z(K)-Z(K-1)                                           PL201600
      UBARK = UBARS/ZSUM                                                PL201700
      ZM = (A1/(UBARK*DPDZ))**B1                                        PL201800
      IF (ZM .LE. Z(K)) GO TO 30                                        PL201900
      K = K+1                                                           PL202000
      IF (K .GT. NZS) GO TO 80                                          PL202100
      GO TO 20                                                          PL202200
   30 IF (Z(K)-ZM .LE. 10.0) GO TO 70                                   PL202300
      IF (DPDZ-3.322E-4) 35,70,35                                       PL202400
   35 CONTINUE                                                          PL202500
      UBARK = UBARS-(Z(K)-Z(K-1))*(UBAR(K)+UBAR(K-1))*0.5               PL202600
      ZBARK = ZSUM-(Z(K)-Z(K-1))                                        PL202700
      ZP = Z(K)                                                         PL202800
   40 ZP = ZP-10.0                                                      PL202900
      IF (ZP .LT. Z(1)) GO TO 85                                        PL203000
      TVP = TV(K)-TPZ(Z(K),ZP,TV(K),TV(K-1),Z(K-1))                     PL203100
      CALL LEAST(Z,TV,DPDZ,K-1,1,ZP,TVP)                                PL203200
      IF (DPDZ .GT. 3.322E-4) GO TO 60                                  PL203300
      DPDZ = 3.322E-4                                                   PL203400
   50 ZM = ZP                                                           PL203500
      GO TO 70                                                          PL203600
   60 UBARZ = UBAR(K)-TPZ(Z(K),ZP,UBAR(K),UBAR(K-1),Z(K-1))             PL203700
      UBARZ = (UBARK+(ZP-Z(K-1))*(UBARZ+UBAR(K-1))*0.5)/(ZBARK+ZP-Z(K-1)PL203800
     1)                                                                 PL203900
      ZM = (A1/(UBARZ*DPDZ))**B1                                        PL204000
```

```
      IF (ZM .GT. ZP) GO TO 50                                          PL204100
      IF (ZM .GT. ZP-10.0) GO TO 70                                     PL204200
      IF (ZP .GE. Z(K-1)) GO TO 40                                      PL204300
      ZM = Z(K-1)                                                       PL204400
C     RETURN ZM AND DPLZ FOR CONTINUOUS SOURCE                          PL204500
   70 IFLG = 0                                                          PL204600
      GO TO 90                                                          PL204700
C     CANNOT CALCULATE ZM AND DPDZ                                      PL204800
   80 IFLG = 1                                                          PL204900
      GO TO 90                                                          PL205000
   85 IFLG = 2                                                          PL205100
   90 RETURN                                                            PL205200
      END                                                               PL205300
```

```
      SUBROUTINE DELTXY                                                  DXY00100
      COMMON /PLUME/QC,A,B,HEAT,RHO,CP,PI,GAMMA,T(21),P(21),Z(21),UBAR( DXY00200
     121),NZS,Q(20),QT,HM,SIGEP(21),SIGAP(21),G,TAUK,NORMAL             DXY00300
     2,TV(21),RH(21),NAMCAS(12),NAMT(12),SIGAR                           DXY00400
      COMMON /REST/ ZM,DPDZ,K,A1,B1,PHI1,ZP,TZ,PZ,PHI,IFLG,KS            DXY00500
      COMMON /OUT/ QF(20,4),WD(21),ISKIP(10),NYS,NDI,NCI,NBK,NPTS,ZRK,   DXY00600
     1JBOT,JTOP,IZMOD,CI(10),DI(10),ZZL(2)                               DXY00700
      COMMON/DISPL/ DXX,DX(21),DYY,DY(21),ILXY                           DXY00800
      IP = 4                                                             DXY00900
      L = 3                                                              DXY01000
      IF (NORMAL .EQ. 1) GO TO 5                                         DXY01100
      IP = 3                                                             DXY01200
      L = 2                                                              DXY01300
    5 UF = 0.0                                                           DXY01400
      ZF = 0.0                                                           DXY01500
      A1 = RHO*CP*PI*GAMMA**L/(3.0*QC*HEAT)                              DXY01600
      IF (NORMAL .EQ. 1) A1 = A1/A                                       DXY01700
      B1 = G/T(1)                                                        DXY01800
      S = 1.0/SQRT(G*DPDZ/T(1))                                          DXY01900
      PPI = PI*5.5555555E-3                                              DXY02000
      TSTR = PI*S                                                        DXY02100
      PPII = 1.0/PPI                                                     DXY02200
      DXX = 0.0                                                          DXY02300
      DYY = 0.0                                                          DXY02400
      I = 0                                                              DXY02500
      TT = 0.0                                                           DXY02600
   10 I = I+1                                                            DXY02700
      IF (I .GE. NZS) GO TO 30                                           DXY02800
      CALL LEAST(Z,TV,DPDZS,I+1,0,0.0,0.0)                               DXY02900
      IF (DPDZS .LT. 3.322E-4) DPDZS = 3.322E-4                          DXY03000
      BK = A1*DPDZS                                                      DXY03100
      IF (NORMAL .EQ. 0) GO TO 12                                        DXY03200
      BK = BK/(Z(I+1)**B)                                                DXY03300
      GO TO 15                                                           DXY03400
   12 CONTINUE                                                           DXY03500
      UFS= UF+(Z(I+1)-Z(I))*.5*(UBAR(I+1)+UBAR(I))                       DXY03600
      ZFS= ZF+(Z(I+1)-Z(I))                                              DXY03700
      BK = BK*UFS/ZFS                                                    DXY03800
   15 CONTINUE                                                           DXY03900
      ZD = BK*Z(I+1)**IP                                                 DXY04000
```

```
      IF (ZD .GT. 2.0) GO TO 20                                        DXY04100
      THETAK = (WD(I+1)+WD(I))*0.5                                     DXY04200
      IF (ABS(WD(I+1)-WD(I)) .GT. 180.0) THETAK = THETAK-180.0         DXY04300
      BB = 1.0-ZD                                                      DXY04400
      IF (BB .GT. 1.0) BB = 1.0                                        DXY04500
      IF (BB .LT.-1.0) BB = -1.0                                       DXY04600
      S = 1.0/SQRT(B1*DPDZS)                                           DXY04700
      TK = S*ARCOS(BB)-TT                                              DXY04800
      TT = TK+TT                                                       DXY04900
      IF (TT .LE. TSTR) GO TO 17                                       DXY05000
      TT = TT-TK                                                       DXY05100
      GO TO 20                                                         DXY05200
   17 CONTINUE                                                         DXY05300
      UF = UFS                                                         DXY05400
      ZF = ZFS                                                         DXY05500
      IF (NORMAL .EQ. 0) GO TO 18                                      DXY05600
      RK = 0.5*(UBAR(I+1)+UBAR(I))*TK                                  DXY05700
      GO TO 19                                                         DXY05800
   18 RK = UF*TK/ZF                                                    DXY05900
   19 CONTINUE                                                         DXY06000
      BB = THETAK*PPI                                                  DXY06100
      DY(I) = DY(I-1)-RK*COS(BB)                                       DXY06200
      DX(I) = DX(I-1)-RK*SIN(BB)                                       DXY06300
      ILXY = I                                                         DXY06400
      GO TO 10                                                         DXY06500
   20 RK = ((ZM-Z(I))/(Z(I+1)-Z(I))*0.5*(UBAR(I+1)-UBAR(I))+UBAR(I))   DXY06600
      IF (NORMAL .EQ. 1) GO TO 25                                      DXY06700
      RK = RK*(ZM-Z(I))+UF                                             DXY06800
      ZF = ZF+(ZM-Z(I))                                                DXY06900
      RK = RK/ZF                                                       DXY07000
   25 RK = RK*(TSTR-TT)                                                DXY07100
      BB = WD(I+1)-WD(I)                                               DXY07200
      IF (BB .GT. 180.0) BB = BB-360.0                                 DXY07300
      IF (BB .LT. -180.0) BB = BB+360.0                                DXY07400
      BB = AMOD(BB,360.0)                                              DXY07500
      THETAM = BB/(Z(I+1)-Z(I))*(ZM-Z(I))+WD(I)                        DXY07600
      THETAK = 0.5*(THETAM+WD(I))                                      DXY07700
      IF (ABS(THETAM-WD(I)) .GT. 180.0) THETAK = THETAK-180.0          DXY07800
      BB = THETAK*PPI                                                  DXY07900
      DX(I) = DX(I-1)-RK*SIN(BB)                                       DXY08000
      DY(I) = DY(I-1)-RK*COS(BB)                                       DXY08100
```

```
      ILXY = I                                                          DXY08200
   28 I = I+1                                                           DXY08300
      IF (I .GE. NZS) GO TO 30                                          DXY08400
      RK = TSTR*0.5*(UBAR(I+1)+UBAR(I))                                 DXY08500
      ZF = 0.5*(WD(I+1)+WD(I))                                          DXY08600
      IF (ABS(WD(I+1)-WD(I)) .GT. 180.0) ZF = ZF-180.0                  DXY08700
      BB = ZF*PPI                                                       DXY08800
      DX(I) = -RK*SIN(BB)                                               DXY08900
      DY(I) = -RK*COS(BB)                                               DXY09000
      GO TO 28                                                          DXY09100
   30 CONTINUE                                                          DXY09200
      I = NZS-1                                                         DXY09300
      DO 80 J =1,I                                                      DXY09400
      IF (DX(J)) 50,40,50                                               DXY09500
   40 IF (DY(J)) 50,80,50                                               DXY09600
   50 BB = 270.0-ATAN2(DY(J),DX(J))*PPII                                DXY09700
      IF (BB .GT. 360.0) BB = BB-360.0                                  DXY09800
      IF (BB .GT. 180.0) GO TO 60                                       DXY09900
      BB = BB+180.0                                                     DXY10000
      GO TO 70                                                          DXY10100
   60 BB = BB-180.0                                                     DXY10200
   70 DX(J) = SQRT(DX(J)*DX(J)+DY(J)*DY(J))                             DXY10300
      DY(J) = BB                                                        DXY10400
   80 CONTINUE                                                          DXY10500
      RETURN                                                            DXY10600
      END                                                               DXY10700
```

```
      SUBROUTINE LEAST(Z,TV,DPDZ,K,ISW,ZP,TVP)
      DIMENSION Z(1),TV(1)
      IF (K .LE. 1) GO TO 30
      L = K
      TVB = 0.0
      ZB = 0.0
      DO 10 I=1,K
      TVB = TVB+TV(I)
   10 ZB = ZB+Z(I)
      IF (ISW .EQ. 0) GO TO 15
      TVB = TVB+TVP
      ZB = ZB+ZP
      L = L+1
   15 TVB = TVB/FLOAT(L)
      ZB = ZB/FLOAT(L)
      S1 = 0.0
      S2 = 0.0
      DO 20 I=1,K
      S1 = S1+(Z(I)-ZB)*(TV(I)-TVB)
   20 S2 = S2+(Z(I)-ZB)**2
      IF (ISW .EQ. 0) GO TO 25
      S1 = S1+(ZP-ZB)*(TVP-TVB)
      S2 = S2+(ZP-ZB)**2
   25 DPDZ = S1/S2
   30 CONTINUE
      RETURN
      END
```

```
      FUNCTION ARCOS(A)                                               ACS00100
      ARCOS = ACOS(A)                                                 ACS00200
      RETURN                                                          ACS00300
      END                                                             ACS00400
```

C.2 FORTRAN SOURCE LISTING FOR THE NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM - VERSION 5

This section contains the complete FORTRAN source listing of the NASA/MSFC Multilayer Models Program - Version 5.

```
C     ***********************************************************************MDL00100
C     *    NASA/MSFC MULTILAYER DIFFUSION MODEL, VERSION V                  *MDL00200
C     ***********************************************************************MDL00300
C                                                                            MDL00400
C                                                                            MDL00500
C                                                                            MDL00600
C                                                                            MDL00700
C       ISKIP= PROGRAM CONTROL OPTIONS                                       MDL00800
C       H = CLOUD STABILIZATION HEIGHT (METERS)                              MDL00900
C       Z     = BOUNDARY HEIGHTS OF LAYERS (METERS)                          MDL01000
C       Q = SOURCE STRENGTH IN LAYER                                         MDL01100
C       UBAR = CALCULATED TRANSPORT SPEED IN LAYER                           MDL01200
C       ALPHA = LATERAL POWER LAW EXPANSION COEFFICIENT                      MDL01300
C       BETA  = VERTICAL POWER LAW EXPANSION COEFFICIENT                     MDL01400
C       SIGYO = STANDARD DEVIATION OF THE LATERAL SOURCE DIMENSION (METER)MDL01500
C       SIGAP = CALCULATED LATERAL DIFFUSION COEFFICIENT IN LAYER            MDL01600
C       SIGXO = STANDARD DEVIATION OF THE ALONG WIND SOURCE DIMENSION        MDL01700
C               (METERS)                                                     MDL01800
C       DELTHP = CALCULATED WIND DIRECTION SHEAR IN LAYER                    MDL01900
C       SIGZO  = STANDARD DEVIATION OF THE VERTICAL SOURCE DIMENSION         MDL02000
C               (METERS)                                                     MDL02100
C       SIGEP = CALCULATED VERTICAL DIFFUSION COEFFICIENT                    MDL02200
C       DELX = RANGE TO SOURCE IN LAYER RELATIVE TO ORIGIN OF REFERENCE      MDL02300
C              GRID SYSTEM (METERS)                                          MDL02400
C       DELY = AZIMUTH BEARING FROM 0 DEGREES NORTH TO SOURCE IN LAYER       MDL02500
C              (DEGREES)                                                     MDL02600
C       THETA = CALCULATED MEAN WIND DIRECTION IN LAYER                      MDL02700
C       IZMOD = MODEL OR MODELS TO USE IN LAYER                              MDL02800
C       DELU  = CALCULATED WIND SPEED SHEAR                                  MDL02900
C       ZZL   = CALCULATION HEIGHTS IN LAYER                                 MDL03000
C       DOS   = CALCULATED VALUE OF DOSAGE                                   MDL03100
C       CON   = CALCULATED VALUE OF CONCENTRATION                            MDL03200
C       PEAKD = PART OF DOSAGE EQUATION                                      MDL03300
C       XX = RANGE TO CALCULATION POINT OF THE POLAR COORDINATE REFERENCE MDL03400
C            GRID SYSTEM (METERS)                                            MDL03500
C       YY = AZIMUTH BEARING FROM 0 DEGREES NORTH TO CALCULATION POINT OF MDL03600
C            THE POLAR COORDINATE REFERENCE GRID                             MDL03700
C       LAT   = LATERAL TERM OF DOSAGE EQUATION                              MDL03800
C       VER   = VERTICAL TERM OF DOSAGE EQUATION                             MDL03900
C       VREF  = REFLECTION TERM OF DOSAGE EQUATION                           MDL04000
```

```
C     T      = SOURCE EMISSION TIME IN LAYER FOR GRAVITATIONAL DEP. (SEC)MDL04100
C     TESTNO = METEOROLOGICAL CASE INFORMATION                         MDL04200
C      DI = DOSAGE ISOPLETH VALUES OF INTEREST                         MDL04300
C      CI  = CONCENTRATION ISOPLETH VALUES OF INTEREST                 MDL04400
C     TI = TIME MEAN CONCENTRATION VALUES OF INTEREST FOR ISOPLETHS    MDL04500
C     SIGZ = CALCULATED STANDARD DEVIATION OF THE VERTICAL DOSAGE      MDL04600
C            DISTRIBUTION                                              MDL04700
C     SIGY = CALCULATED STANDARD DEVIATION OF THE LATERAL DOSAGE       MDL04800
C            DISTRIBUTION                                              MDL04900
C     SIGX = CALCULATED STANDARD DEVIATION OF THE ALONG WIND DOSAGE    MDL05000
C            DISTRIBUTION                                              MDL05100
C     SQR2P =  SQUARE ROOT TWO PI                                      MDL05200
C     L      = LENGTH OF CLOUD IN ALONG WIND DIRECTION                 MDL05300
C     I      = INDEX OF X COORDINATES                                  MDL05400
C     J      = INDEX OF Y COORDINATES                                  MDL05500
C     KK     = INDEX OF LAYERS                                         MDL05600
C     K = INDEX OVER CALCULATION HEIGHTS ZZL                           MDL05700
C     STO1  = TEMP STORAGE                                             MDL05800
C     STO2  = TEMP STORAGE                                             MDL05900
C     STO3  = TEMP STORAGE                                             MDL06000
C     TAST  = TIME OF LAYER STRUCTURE CHANGE (SECONDS)                 MDL06100
C     NBK = NO OF DISTINCT GROUPS OF LAYERS THAT FORM INTO ONE AT TIME MDL06200
C            TAST.                                                     MDL06300
C     ILK   = INDEX ON NEW LAYERS AFTER TIME TAST                      MDL06400
C     NXS   = NO OF X COORDINATES                                      MDL06500
C     NYS   = NO OF Y COORDINATES                                      MDL06600
C     NZS   = NO OF LAYER BOUNDARIES                                   MDL06700
C     NDI   = NO OF DOSAGE ISOPLETHS                                   MDL06800
C     NCI   = NO OF CONCENTRATION ISOPLETHS                            MDL06900
C     NT1 = NO OF TIME MEAN CONCENTRATION ISOPLETHS                    MDL07000
C     NPTS = NO OF CALCULATION HEIGHTS ZZL                             MDL07100
C     RAD   = PI/180                                                   MDL07200
C     NNZ   = NZS-1 NO OF LAYERS                                       MDL07300
C     ITOP   = TOP OF NEW LAYER AFTER TAST IN TERMS OF OLD LAYER STRUCTURMDL07400
C     IBOT   = BOTTOM OF NEW LAYER AFTER TAST IN TERMS OF OLD LAYER    MDL07500
C            STRUCTURE (ITOP AND IBOT INDEXES)                         MDL07600
C     XAST   = CALCULATE DISTANCE TO TAST                              MDL07700
C     SIGXNK = SIGX OF NEW LAYER STRUCTURE                             MDL07800
C     LAMBDA = WASHOUT COEFFICIENT                                     MDL07900
C     TIM1 = TIME OF START OF RAIN (SECONDS)                           MDL08000
C     ZLIM   = MAXIMUM HEIGHT OF WASHOUT                               MDL08100
```

```
C      WASHOU = CALCULATE WASHOUT AT GROUND                                      MDL08200
C      UBARK = WIND SPEED AT EACH LAYER BOUNDARY, LOWER BOUNDARY OF LAYERMDL08300
C              1 FOR UBARK IS ASSUMED AT ZRK  (METERS/SEC)                        MDL08400
C      SIGAK = SIGAP (INITIAL) AT EACH LAYER BOUNDARY, LOWER BOUNDARY OF MDL08500
C              LAYER 1 FOR SIGAK IS ASSUMED AT ZRK (DEGREES)                      MDL08600
C      SIGEK = SIGEP (INITIAL) AT EACH LAYER BOUNDARY, LOWER BOUNDARY OF MDL08700
C              LAYER 1 FOR SIGEK IS ASSUMED AT ZRK (DEGREES)                      MDL08800
C      ZRK = REFERENCE HEIGHT IN SURFACE LAYER (METERS)                           MDL08900
C      THETAK = WIND DIRECTION AT LAYER BOUNDARIES (DEGREES)                      MDL09000
C      TAUK  = TIME IN SECONDS REQUIRED FOR LATERAL CLOUD STABILIZATION  MDL09100
C      TAUOK = SAMPLING PERIOD IN SECONDS AT THE TOP OF THE LAYER         MDL09200
C      DECAY = DECAY COEFFICIENT IN DOSAGE EQUATION                               MDL09300
C      UBARL = WIND SPEED AT BOTTOM AND TOP OF EACH NEW LAYER AFTER LAYERMDL09400
C              CHANGE  (METERS/SEC)                                               MDL09500
C      SIGAL = SIGAP AT BOTTOM AND TOP OF EACH NEW LAYER AFTER LAYER     MDL09600
C              CHANGE  (DEGREES)                                                  MDL09700
C      SIGEL = SIGEP AT BOTTOM AND TOP OF EACH NEW LAYER AFTER LAYER     MDL09800
C              CHANGE  (DEGREES)                                                  MDL09900
C      ZRL = REFERENCE HEIGHT IN SURFACE LAYER OF NEW STRUCTURE (METERS) MDL10000
C      THETAL = WIND DIRECTION AT BOTTOM AND TOP OF EACH NEW LAYER AFTER MDL10100
C      TAUL  = TIME IN SECONDS FOR LATERAL CLOUD STABILIZATION IN NEW    MDL10200
C              LAYER STRUCTURE                                                    MDL10300
C      TAUOL = TIME IN SECONDS OF SAMPLING PERIOD AT TOP OF NEW LAYER    MDL10400
C      JBOT  = INPUT LAYER NUMBER OF BOTTOM OF NEW LAYER STRUCTURE        MDL10500
C              RELATIVE TO OLD                                                    MDL10600
C      JTOP  = INPUT LAYER NUMBER OF TOP OF NEW LAYER STRUCTURE           MDL10700
C              RELATIVE TO OLD                                                    MDL10800
C      VS    = SETTLING VELOCITY IN GRAVITATIONAL DEPOSITION MODEL         MDL10900
C      PERC  = FREQUENCY OF VS                                                    MDL11000
C      ACCUR = DESIRED ACCURACY COEFFICIENT (.45) INSURES THAT GROUND    MDL11100
C              DEPOSITION FROM NXCI POINT SOURCES IN THE LAYER VARIES    MDL11200
C              LESS THAN TEN PERCENT FROM DEPOSITION EXPECTED FROM A     MDL11300
C              VERTICAL LINE SOURCE IN THE LAYER, FOR (.32) REDUCED TO   MDL11400
C              FIVE PERCENT                                                       MDL11500
C      VB    = SETTLING VELOCITIES FROM A BURST OR DESTRUCT IN LAYER NNZ MDL11600
C      PERCB = FREQUENCY OF VB                                                    MDL11700
C      HB    = HEIGHT OF BURST (METERS)                                           MDL11800
C      PPWR  = CALCULATED WIND SPEED POWER LAW EXPONENT                          MDL11900
C      QPWR  = CALCULATED SIGEP POWER LAW EXPONENT                               MDL12000
C      MPWR  = CALCULATED SIGAP POWER LAW EXPONENT                               MDL12100
C      DTHK  = WIND ANGLE SHEAR                                                   MDL12200
```

```
C      NVS    = NUMBER OF SETTLING VELOCITIES VS                             MDL12300
C      NVB    = NUMBER OF SETTLING VELOCITIES VB                             MDL12400
C      II     = INDEX ON VS AND VB                                           MDL12500
C      DEP    = TEMP STORAGE                                                 MDL12600
C      YBARY = CALCULATED CORRDINATE OF POINT ON CLOUD AXIS OF VS AT         MDL12700
C              INTERSECTION WITH GROUND  (DEPOSITION)                        MDL12800
C      XBARX = CALCULATED CORRDINATE OF POINT ON CLOUD AXIS OF VS AT         MDL12900
C              INTERSECTION WITH GROUND  (DEPOSITION)                        MDL13000
C      UBARNK = CALCULATED WIND SPEED  (DEPOSITION)                          MDL13100
C      BETANK = CALCULATED BETA   (DEPOSITION)                               MDL13200
C      ALPHNK = CALCULATED ALPHA  (DEPOSITION)                               MDL13300
C      SQBAR  = TEMP STORAGE                                                 MDL13400
C      ANG    =  ANGLE  TO POINT XBARX,YBARY  (DEPOSITION)                   MDL13500
C      NXCI   = NUMBER OF POINT SOURCES IN LAYER  (DEPOSITION)               MDL13600
C      DEPN   = CALCULATED VALUE OF GRAVITATIONAL DEPOSITION                 MDL13700
C      SIGYNK = SIGY OF NEW LAYER STRUCTURE IN CALCULATION OF DOSAGE AND MDL13800
C               CONCENTRATION                                                MDL13900
C      SIGENK = CALCULATED SIGEP (DEPOSITION)                                MDL14000
C      SIGANK = CALCULATED SIGAP (DEPOSITION)                                MDL14100
C      TIMAV = CONCENTRATION AVERAGING TIME (SECONDS)                        MDL14200
C      AVCON = AVERAGE CONCENTRATION                                         MDL14300
C      PASSTM = TIME OF CLOUD PASSAGE                                        MDL14400
C      AVMXCN = MAXIMUM AVERAGE CONCENTRATION                                MDL14500
C      XRY    = DISTANCE DOWNWIND FROM THE VIRTUAL POINT SOURCE OVER         MDL14600
C              WHICH RECTILINEAR EXPANSION OCCURS LATERALLY   (METERS)       MDL14700
C      XRZ    = DISTANCE DOWNWIND FROM THE VIRTUAL POINT SOURCE OVER         MDL14800
C              WHICH RECTILINEAR EXPANSION OCCURS VERTICALLY  (METERS)       MDL14900
C      XLRY   = DISTANCE FROM TRUE SOURCE TO POINT OF MEASUREMENT OF         MDL15000
C              SIGYO  (METERS)                                               MDL15100
C      XLRZ   = DISTANCE FROM TRUE SOURCE TO POINT OF MEASUREMENT OF         MDL15200
C              SIGZO  (METERS)                                               MDL15300
C      GAMMA  = FRACTION OF MATERIAL REFLECTED AT THE SURFACE (=1 FOR        MDL15400
C               COMPLETE REFLECTION, =0 FOR NO REFLECTION)                   MDL15500
C      GAMMAP = 1.0-GAMMA                                                    MDL15600
C      NAMCAS = SPECIAL CASE IDENTIFICATION INFORMATION                      MDL15700
C      SCL - MAP SCALE FACTOR IN INCHES FOR ISOPLETH PLOTS. IF THE MAP       MDL15800
C            SCALE FACTOR IS 1 INCH = 24000 INCHES THEN SCL WOULD BE         MDL15900
C            INPUT AS 24000. (IF 0 THE PROGRAM WILL CALCULATE SCL)           MDL16000
C      ISW - SWITCH FOR MAXIMUM CENTERLINE PLOTS. IF SET TO 0 OR 2           MDL16100
C            LOG-LOG SCALING IS USED. IF SET TO 1 LINEAR IS USED.            MDL16200
C      XMAXJN - MAXIMUM ALONGWIND DISTANCE FROM THE LAUNCH SITE FOR          MDL16300
```

```
C                  MAXIMUM CENTERLINE PLOTS (METERS) (IF 0 PROG CALCULATES) MDL16400
C        XMAXIN - MAXIMUM ALONGWIND DISTANCE FROM THE LAUNCH SITE FOR      MDL16500
C                   ISOPLETHS (METERS) (IF 0 PROGRAM CALCULATES)           MDL16600
C        YMAXIN - MAXIMUM CROSSWIND DISTANCE FOR ISOPLETHS (METERS) (IF 0  MDL16700
C                  PROGRAM CALCULATES)                                     MDL16800
C        YMAXJN - MAXIMUM NUMBER OF LOG CYCLES FOR THE VERTICAL AXIS OF    MDL16900
C                  THE MAXIMUM CENTERLINE PLOTS IF ISW = 0 OR 2. OR. MAXIMUMMDL17000
C                  VALUE OF THE VERTICAL AXIS IF ISW = 1. (IF 0 PROGRAM    MDL17100
C                  CALCULATES)                                             MDL17200
C        TEMPK - VIRTUAL POTENTIAL TEMPERATURE AT EACH LAYER BOUNDARY. THISMDL17300
C                 ARRAY IS USED TO SEE IF THERE IS A NEGATIVE LAPSE RATE   MDL17400
C                 IN THE LAYER. THE PROG CHECKS TO SEE IF THE WIND SPEED   MDL17500
C                 SHEAR IS NEGATIVE. IF IT IS AND ALSO THE LAPSE RATE IS   MDL17600
C                 NEGATIVE THE PROGRAM USES THE ABSOLUTE VALUE OF THE SPEED MDL17700
C                 SHEAR . IF THE SPEED SHEAR IS NEGATIVE AND THE LAPSE RATE MDL17800
C                 IS POSITIVE OR TEMPK IS NOT INPUT THE PROGRAM USES 0 WIND MDL17900
C                 SPEED SHEAR.                                             MDL18000
C        TEMPL - VIRTUAL POTENTIAL TEMPERATURE AT EACH LAYER BOUNDARY OF   MDL18100
C                 THE NEW LAYER STRUCTURE.                                 MDL18200
C                                                                          MDL18300
C                                                                          MDL18400
C                    - PROGRAM INPUT PARAMETERS -                          MDL18500
C        ISKIP, H, Z, Q, ALPHA, BETA, SIGYO, SIGZO, SIGXO, DELTHP, DELX,   MDL18600
C        DELY, IZMOD, ZZL, XX, YY, T, TESTNO, DI, CI, TI, TAST,            MDL18700
C        NXS,NYS,NZS,NDI, NCI, NTI, TIM1, UBARK, SIGAK, ZLIM,              MDL18800
C        SIGEK, ZRK, THETAK, TAUK, TAUOU, DECAY, UBARL, SIGAL, SIGEL, ZRL, MDL18900
C        THETAL, TAUL, TAUOL, VS, PERC, ACCUR, VB, PERCB, HB,NAMCAS        MDL19000
C        NVS, NVB, NPTS, TIMAV, LAMBDA (BLAMDA), XRY, XRZ, XLRY, XLRZ      MDL19100
C           SOME OF THE ABOVE PARAMETERS ARE AUTOMATICALLY DETERMINED BY   MDL19200
C           THE PROGRAM, CONSULT THE PROGRAM DOCUMENTATION TO DETERMINE    MDL19300
C           WHICH THEY ARE. ALL INPUTS ARE READ VIA THE FORTRAN NAMELISTS  MDL19400
C           NAM2 IN SUBROUTINE READER                                      MDL19500
C                                                                          MDL19600
C                                                                          MDL19700
C                                                                          MDL19800
C                                                                          MDL19900
         COMMON /PARAMT/ TESTNO(12),       ISKIP(15),NXS,NYS,NZS,NDI,NCI,  MDL20000
        1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     MDL20100
        2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      MDL20200
        3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     MDL20300
        4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), MDL20400
```

```
      5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),MDL20500
      6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)MDL20600
      7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  MDL20700
       COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),MDL20800
      1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SGR2P,L,TH,I,J,KK,STO1,    MDL20900
      2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, MDL21000
      3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    MDL21100
      4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          MDL21200
      5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       MDL21300
      6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   MDL21400
      7NYSS,CDAMX(3)                                                     MDL21500
       DIMENSION CON(1),DOS(1),AVCON(1),PASSTM(1)                        MDL21600
       EQUIVALENCE (CON,DEPN),(DOS,DEPN(1,2)),(AVCON,DEPN(1,3)),(PASSTM,DMDL21700
      1EPN(1,4))                                                         MDL21800
       REAL MPWR,L,LAT,LAMBDA                                            MDL21900
       INTEGER TESTNO                                                    MDL22000
C                  *** INPUT SECTION ***                                 MDL22100
       SQR2P = 2.5066283                                                 MDL22200
       RAD = .01745329                                                   MDL22300
       IFF = 1                                                           MDL22400
       MBR = 0                                                           MDL22500
C                            READ MODEL PARAMETERS                       MDL22600
     1 CALL READER(IFF)                                                  MDL22700
       IFF = 2                                                           MDL22800
       IF (KSW(2) .LE. 0) GO TO 5                                        MDL22900
C                      EXECUTE GRAVITATIONAL DEPOSITION MODEL            MDL23000
       CALL DEPOS                                                        MDL23100
       GO TO 700                                                         MDL23200
     5 CONTINUE                                                          MDL23300
       IF (ISKIP(2) .LE. 1.AND.ISKIP(3) .LE. 1) GO TO 6                  MDL23400
       IF (MBR .EQ. 5) GO TO 6                                           MDL23500
       MBR = 5                                                           MDL23600
       CALL IDENT(35,'HARD COPY, 1 EACH, PLUS FILM')                     MDL23700
       CALL SETMIV(0,0,0,0)                                              MDL23800
     6 CONTINUE                                                          MDL23900
       DO 8 I=1,3                                                        MDL24000
     8 CDAMX(I) = 0.0                                                    MDL24100
       ILK = 1                                                           MDL24200
       DO 10 J=1,41                                                      MDL24300
       DO 10 I=1,41                                                      MDL24400
    10 DEPN(I,J) = 0.0                                                   MDL24500
```

```
   20 CONTINUE                                                          MDL24600
      KTK = 1                                                           MDL24700
      K = 1                                                             MDL24800
      NYSS = NYS                                                        MDL24900
      IMB = 0                                                           MDL25000
      IFG = 0                                                           MDL25100
      DO 500 KK=1,NNZ                                                   MDL25200
C                    *** LIST INPUT PARAMETERS ***                      MDL25300
      WRITE (6,903) KK                                                  MDL25400
      WRITE (6,904)                                                     MDL25500
      IF (KK .NE. 1) GO TO 92                                           MDL25600
      WRITE (6,905) Q(KK),ZRK,UBARK(KK),UBARK(KK+1),SIGAK(KK),SIGAK(KK+1MDL25700
     1),SIGEK(KK),SIGEK(KK+1),TAUK,TAUOK,SIGXO(KK),SIGYO(KK),SIGZO(KK), MDL25800
     2THETAK(KK),THETAK(KK+1),Z(KK),ALPHA(KK),BETA(KK),H,DELX(KK),      MDL25900
     3DELY(KK),IZMOD(KK),TIM1,ZLIM,LAMBDA,TIMAV,XRY,XRZ,XLRY,XLRZ       MDL26000
     4,GAMMAP(1)                                                        MDL26100
      GO TO 93                                                          MDL26200
   92 CONTINUE                                                          MDL26300
      WRITE (6,918) Q(KK),UBARK(KK),UBARK(KK+1),SIGAK(KK),SIGAK(KK+1),  MDL26400
     1SIGEK(KK),SIGEK(KK+1),SIGXO(KK),SIGYO(KK),SIGZO(KK),THETAK(KK),   MDL26500
     2THETAK(KK+1),Z(KK),ALPHA(KK),BETA(KK),DELX(KK),DELY(KK),          MDL26600
     3IZMOD(KK)                                                         MDL26700
   93 IF (KK .NE. NNZ) GO TO 94                                         MDL26800
      WRITE (6,919) Z(KK+1)                                             MDL26900
   94 CONTINUE                                                          MDL27000
      NNZILK = NNZ+ILK                                                  MDL27100
      IF (NBK .EQ. 0.OR.KK .NE. JBOT(ILK)) GO TO 97                     MDL27200
      IF (JBOT(ILK) .NE. 1) GO TO 96                                    MDL27300
      LSP = ILK*2-1                                                     MDL27400
      WRITE (6,920) ZRL,UBARL(LSP),UBARL(LSP+1),SIGAL(LSP),SIGAL(LSP+1),MDL27500
     1SIGEL(LSP),SIGEL(LSP+1),THETAL(LSP),THETAL(LSP+1),TAUL,TAUOL,     MDL27600
     2ALPHA(NNZILK),BETA(NNZILK),TAST(ILK),JBOT(ILK),JTOP(ILK)          MDL27700
      GO TO 97                                                          MDL27800
   96 CONTINUE                                                          MDL27900
      LSP = ILK*2-1                                                     MDL28000
      WRITE (6,921) UBARL(LSP),UBARL(LSP+1),SIGAL(LSP),SIGAL(LSP+1),    MDL28100
     1SIGEL(LSP),SIGEL(LSP+1),THETAL(LSP),THETAL(LSP+1),TAUL,TAUOL,ALPHAMDL28200
     2(NNZILK),BETA(NNZILK),TAST(ILK),JBOT(ILK),JTOP(ILK)               MDL28300
   97 CONTINUE                                                          MDL28400
      WRITE (6,922) UBAR(KK),THETA(KK),DELTHP(KK),DELU(KK),SIGAP(KK),   MDL28500
     1SIGEP(KK)                                                         MDL28600
```

```
      IF (NBK .EQ. 0.OR.KK.NE. JBOT(ILK)) GO TO 98                        MDL28700
      WRITE (6,923) UBAR(NNZILK),THETA(NNZILK),DELTHP(NNZILK),            MDL28800
     1DELU(NNZILK),SIGAP(NNZILK),SIGEP(NNZILK)                            MDL28900
   98 CONTINUE                                                            MDL29000
       CALL TESTR(KTK)                                                    MDL29100
      WRITE (6,917)                                                       MDL29200
C                  *** GENERAL GRID PATTERN CALCULATIONS ***              MDL29300
  140 CONTINUE                                                            MDL29400
      JF = NNZ+ILK-1                                                      MDL29500
      IF (KSW(1) .LE. 0) GO TO 145                                        MDL29600
      IF (IFG .EQ. 1) GO TO 500                                           MDL29700
      GO TO 148                                                           MDL29800
  145 CONTINUE                                                            MDL29900
      IF (K .GT. NPTS) GO TO 500                                          MDL30000
      IF (ZZL(K)-Z(KK+1)) 148,500,500                                     MDL30100
  148 MDLS = MODLS(KK)                                                    MDL30200
      IF (NBK.GT.0.AND.KK.GE.IBOT.AND.KK.LE.ITOP) MDLS = 4                MDL30300
      IF (NBK .LE. 0) GO TO 149                                           MDL30400
      IF (KK .LT. IBOT.OR.KK .GT. ITOP) GO TO 149                         MDL30500
      YT = THETA(JF)+180.0                                                MDL30600
      GO TO 150                                                           MDL30700
  149 YT = THETA(KK)+180.0                                                MDL30800
  150 CONTINUE                                                            MDL30900
C     DEFAULT YY (ANGULAR AXES)                                           MDL31000
      IF (IMB .EQ. 1) GO TO 153                                           MDL31100
      IF (NYS .GT. 0) GO TO 153                                           MDL31200
      DEP = YT*0.2+0.5                                                    MDL31300
      NYSS = DEP                                                          MDL31400
      DEP = 5*NYSS                                                        MDL31500
      NYSS = 41                                                           MDL31600
      DO 152 J=1,NYSS                                                     MDL31700
  152 YY(J) = DEP+YSV(J)                                                  MDL31800
  153 CONTINUE                                                            MDL31900
      DO 200 I=1,NXS                                                      MDL32000
      DO 160 J=1,NYSS                                                     MDL32100
      IF (KSW(1) .GT. 0) GO TO 155                                        MDL32200
      CALL BREAK(K,XX(I),YY(J))                                           MDL32300
      CDAMX(1) = AMAX1(CDAMX(1),CON(J))                                   MDL32400
      CDAMX(2) = AMAX1(CDAMX(2),DOS(J))                                   MDL32500
      CDAMX(3) = AMAX1(CDAMX(3),AVCON(J))                                 MDL32600
      GO TO 160                                                           MDL32700
```

```
  155 CALL WASHT                                                    MDL32800
  160 CONTINUE                                                      MDL32900
      IF (KSW(1) .LE. 0) GO TO 170                                  MDL33000
      IMB = 1                                                       MDL33100
       GO TO 200                                                    MDL33200
  170 CONTINUE                                                      MDL33300
C     OUTPUT GENERAL GRID PATTERN CALCULATIONS                      MDL33400
      KOUT = 4*I-3                                                  MDL33500
      CALL INTOUT(CON,KOUT,NYSS,2,1,1)                              MDL33600
      KOUT = 4*I-2                                                  MDL33700
      CALL INTOUT(DOS,KOUT,NYSS,2,1,1)                              MDL33800
      KOUT = 4*I-1                                                  MDL33900
      CALL INTOUT(AVCON,KOUT,NYSS,2,1,1)                            MDL34000
      KOUT = 4*I                                                    MDL34100
      CALL INTOUT(PASSTM,KOUT,NYSS,2,1,1)                           MDL34200
  200 CONTINUE                                                      MDL34300
      IF (KSW(1) .LE. 0) GO TO 210                                  MDL34400
      IF (Z(KK+1) .LT. ZLIM) GO TO 500                              MDL34500
      IFG = 1                                                       MDL34600
C     OUTPUT WASHOUT DEPOSITION PATTERNS                            MDL34700
      DO 205 J=1,NYSS                                               MDL34800
      DO 205 I=1,NXS                                                MDL34900
  205 CDAMX(1) = AMAX1(CDAMX(1),DEPN(I,J))                          MDL35000
      MDLS = 5                                                      MDL35100
      ZZL(1) = Z(1)                                                 MDL35200
      CALL GENPRT(1)                                                MDL35300
      GO TO 500                                                     MDL35400
  210 CONTINUE                                                      MDL35500
      CALL GENPRT(K)                                                MDL35600
      K = K+1                                                       MDL35700
      IF (K .GT. NPTS) GO TO 500                                    MDL35800
      IF (ZZL(K) .LT. Z(KK+1)) GO TO 148                            MDL35900
  500 CONTINUE                                                      MDL36000
C      **** LOOP FOR NEXT TEST *****                                MDL36100
  700 CONTINUE                                                      MDL36200
      IF (NPS .EQ. 0) GO TO 1                                       MDL36300
  777 CONTINUE                                                      MDL36400
  800 CONTINUE                                                      MDL36500
      IF (MBR .EQ. 5) CALL ENDJOB                                   MDL36600
  903 FORMAT (1H0,55X,11H***** LAYER,I2,6H *****)                   MDL36700
  904 FORMAT (1H0,57X,16H** INPUT DATA **)                          MDL36800
```

```
  905 FORMAT (4H0 Q=,E14.8,6H, ZRK=,F7.3,17H, UBAR AT BOTTOM=,F8.4,14H, MDL36900
     1UBAR AT TOP=,F8.4,18H, SIGAK AT BOTTOM=,F8.5/14H SIGAK AT TOP=,F8.MDL37000
     25,18H, SIGEK AT BOTTOM=,F8.5,15H, SIGEK AT TOP=,F8.5,7H, TAUK=,F8.MDL37100
     33,8H, TAUOK=,F8.3/7H SIGXO=,F9.4,8H, SIGYO=,F9.4,8H, SIGZO=,F9.4,1MDL37200
     49H, THETAK AT BOTTOM=,F8.3,16H, THETAK AT TOP=,F8.3,4H, Z=,F9.3/7HMDL37300
     5 ALPHA=,F4.1,6H BETA=,F4.1,4H, H=,F9.3,7H, DELX=,E14.8,7H, DELY=,EMDL37400
     614.8,8H, IZMOD=,I3,7H, TIM1=,E14.8/6H ZLIM=,F9.3,                  MDL37500
     79H, LAMBDA=,F7.4,8H, TIMAV=,F8.3,6H, XRY=,F8.3,6H, XRZ=,F8.3,7H, XMDL37600
     8LRY=,F8.3,7H, XLRZ=,F8.3,9H, GAMMAP=,F5.3)                         MDL37700
  917 FORMAT (12X,18(6H------)/)                                         MDL37800
  918 FORMAT (4H0 Q=,E14.8,17H, UBAR AT BOTTOM=,F8.4,14H, UBAR AT TOP=,FMDL37900
     18.4,18H, SIGAK AT BOTTOM=,F8.5,15H, SIGAK AT TOP=,F8.5/17H SIGEK AMDL38000
     2T BOTTOM=,F8.5,15H, SIGEK AT TOP=,F8.5,8H, SIGXO=,F9.4,8H, SIGYO=,MDL38100
     3F9.4,8H, SIGZO=,F9.4,19H, THETAK AT BOTTOM=,F8.3/15H THETAK AT TOPMDL38200
     4=,F8.3,4H, Z=,F9.3,8H, ALPHA=,F4.1,7H, BETA=,F4.1,             7H, MDL38300
     5DELX=,E14.8,7H, DELY=,E14.8/7H IZMOD=,I3)                          MDL38400
  919 FORMAT (1X,10H Z AT TOP=,F10.4)                                    MDL38500
  920 FORMAT (6H0 ZRL=,F7.3,18H, UBARL AT BOTTOM=,F8.4,15H, UBARL AT TOPMDL38600
     1=,F8.4,18H, SIGAL AT BOTTOM=,F8.5,15H, SIGAL AT TOP=,F8.5/17H SIGEMDL38700
     2L AT BOTTOM=,F8.5,15H, SIGEL AT TOP=,F8.5,19H, THETAL AT BOTTOM=,FMDL38800
     38.3,16H, THETAL AT TOP=,F8.3,7H, TAUL=,F8.3/7H TAUOL=,F8.3,8H, ALPMDL38900
     4HL=,F4.1,7H, BETL=,F4.1,7H, TAST=,E14.8,7H, JBOT=,I2,7H, JTOP=,I2)MDL39000
  921 FORMAT (18H0 UBARL AT BOTTOM=,F8.4,15H, UBARL AT TOP=,F8.4,18H, SIMDL39100
     1GAL AT BOTTOM=,F8.5,15H, SIGAL AT TOP=,F8.5/17H SIGEL AT BOTTOM=,FMDL39200
     28.5,15H, SIGEL AT TOP=,F8.5,19H, THETAL AT BOTTOM=,F8.3,16H, THETAMDL39300
     3L AT TOP=,F8.3,7H, TAUL=,F8.3/7H TAUOL=,F8.3,8H, ALPHL=,F4.1,7H, BMDL39400
     4ETL=,F4.1,7H, TAST=,E14.8,7H, JBOT=,I2,7H, JTOP=,I2)               MDL39500
  922 FORMAT (1H0,56HCALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UMDL39600
     1BAR =,F10.5,9H, THETA =,F10.5,10H, DELTHP =,F10.5,8H, DELU =,F10.5MDL39700
     2/1X,09H, SIGAP =,F10.5,9H, SIGEP =,F10.5)                          MDL39800
  923 FORMAT (1H0,63HCALCULATED INPUT PARAMETERS FOR LAYER CHANGE MODEL MDL39900
     14 *** UBAR =,F10.5,9H, THETA =,F10.5,10H, DELTHP =,F10.5/1X,8H  DEMDL40000
     2LU =,F10.5,9H, SIGAP =,F10.5,9H, SIGEP =,F10.5)                    MDL40100
      STOP                                                               MDL40200
      END                                                                MDL40300
```

```
      SUBROUTINE BREAK(K,XO,YO)                                         BRK00100
      COMMON /PARAMT/ TESTNO(12),        ISKIP(15),NXS,NYS,NZS,NDI,NCI, BRK00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     BRK00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      BRK00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     BRK00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), BRK00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),BRK00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)BRK00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  BRK00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),BRK01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    BRK01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, BRK01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    BRK01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          BRK01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       BRK01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   BRK01600
     7NYSS,CDAMX(3)                                                     BRK01700
      DIMENSION CON(1),DOS(1),AVCON(1),PASSTM(1),ERFX(1)                BRK01800
      EGUIVALENCE (CON,DEPN),(DOS,DEPN(1,2)),(AVCON,DEPN(1,3)),(PASSTM,DBRK01900
     1EPN(1,4)),(ERFX,ANG(10))                                          BRK02000
      REAL MPWR,L,LAT,LAMBDA                                            BRK02100
      INTEGER TESTNO                                                    BRK02200
C     *** THIS SUBROUTINE CALCULATES DOSAGE,CONCENTRATION AND WASHOUT **BRK02300
C     *** ON A GENERAL GRID WITHIN THE SECTOR DELPHI.                   BRK02400
C     DETERMINE LOCATION OF RECEPTOR RELATIVE TO SOURCE AND WIND        BRK02500
C     DIRECTION                                                         BRK02600
      CALL COORD(N,KK,X,Y,XO,YO,ASP,XS,1)                               BRK02700
      DOS(J) = 0.0                                                      BRK02800
      CON(J) = 0.0                                                      BRK02900
      IF (NBK .NE. 0.AND.IBOT .LE. KK.AND.KK .LE. ITOP) GO TO 135       BRK03000
      IS = 1                                                            BRK03100
      IF (N .EQ. 9) GO TO 310                                           BRK03200
C                         CALCULATION OF MODELS 1,2,3                   BRK03300
  125 CALL SIGMA(X,KK,1)                                                BRK03400
      IF (SIGY) 130,130,126                                             BRK03500
  126 IF (SIGZ .LT. 0.0 .AND. MODLS(KK) .EQ. 3) GO TO 130               BRK03600
      LAT = Y/SIGY                                                      BRK03700
      LAT = -0.5*LAT*LAT                                                BRK03800
      IF (LAT .LT. -60.0) GO TO 130                                     BRK03900
      LAT = EXP(LAT)                                                    BRK04000
```

```
      PEAKD = Q(KK)/(SGR2P*SIGY*UBAR(KK))                                BRK04100
      IF (MODLS(KK) .EQ. 3) GO TO 20                                     BRK04200
      PEAKD = PEAKD/(Z(KK+1)-Z(KK))                                      BRK04300
      GO TO 21                                                           BRK04400
   20 PEAKD = PEAKD/(SGR2P*SIGZ)                                         BRK04500
   21 CONTINUE                                                           BRK04600
      VER = 0.0                                                          BRK04700
      VREF = 1.0                                                         BRK04800
      IF (MODLS(KK) .NE. 3) GO TO 70                                     BRK04900
      VREF = 0.0                                                         BRK05000
      TMPQ1 = -0.5/(SIGZ*SIGZ)                                           BRK05100
      A = H-ZZL(K)                                                       BRK05200
      B = H-Z(KK)-Z(KK)+ZZL(K)                                           BRK05300
      C = B*B                                                            BRK05400
      C = C*TMPQ1                                                        BRK05500
      A1 = Z(KK+1)-Z(KK)                                                 BRK05600
      IF (C .LT. -30.0) GO TO 70                                         BRK05700
      D = A*A                                                            BRK05800
      D = D*TMPQ1                                                        BRK05900
      IF (D .LT. -30.0) GO TO 50                                         BRK06000
      VER = EXP(D)                                                       BRK06100
   50 VER = VER+GAMMA(1)*EXP(C)                                          BRK06200
      C = 1.0                                                            BRK06300
      D = GAMMA(1)                                                       BRK06400
      E = D*D                                                            BRK06500
      AB = 0.0                                                           BRK06600
   60 AB = AB+2.0                                                        BRK06700
      TR = AB*A1                                                         BRK06800
      TLIM = TR-B                                                        BRK06900
      TLIM = TLIM*TLIM*TMPQ1                                             BRK07000
      IF (TLIM .LT. -10.0) GO TO 70                                      BRK07100
      STO1 = TR+A                                                        BRK07200
      STO2 = TR-A                                                        BRK07300
      STO3 = TR+B                                                        BRK07400
      VREF = VREF+C*EXP(TLIM)+D*(EXP(STO1*STO1*TMPQ1)+EXP(STO2*STO2*TMPQBRK07500
     11))+E*EXP(STO3*STO3*TMPQ1)                                         BRK07600
      C = D                                                              BRK07700
      D = E                                                              BRK07800
      E = E*GAMMA(1)                                                     BRK07900
      GO TO 60                                                           BRK08000
   70 CONTINUE                                                           BRK08100
```

```
      TMPQ1 = X/UBAR(KK)                                                BRK08200
      DOS(J) = PEAKD*LAT*(VER+VREF)                                     BRK08300
      IF (DECAY .GT. 0.0) DOS(J) = DOS(J)*EXP(-DECAY*TMPQ1)             BRK08400
      IF (LAMBDA .LE. 0.0.OR.TIM1 .GE. TMPQ1) GO TO 127                 BRK08500
      IF (Z(KK) .GT. ZLIM) GO TO 127                                    BRK08600
      AB = EXP(-LAMBDA*(TMPQ1-TIM1))                                    BRK08700
      DOS(J) = DOS(J)*AB                                                BRK08800
  127 CONTINUE                                                          BRK08900
      ANG(1) = UBAR(KK)                                                 BRK09000
      ANG(2) = SIGX                                                     BRK09100
      IF (SIGX) 129,129,128                                             BRK09200
  128 CONTINUE                                                          BRK09300
      CON(J) = DOS(J)*UBAR(KK)/(SQR2P*SIGX)                             BRK09400
  129 CONTINUE                                                          BRK09500
  130 IF (IS .EQ. 1) GO TO 310                                          BRK09600
      GO TO 140                                                         BRK09700
  135 IS = 0                                                            BRK09800
      IF (N .NE. 9) GO TO 125                                           BRK09900
C     CALCULATION OF THE FULL TRANSITION MODEL, MODEL4                  BRK10000
  140 DO 200 M=IBOT,ITOP                                                BRK10100
      CALL COORD(N,M,X,Y,XO,YO,ASP,XS,2)                                BRK10200
      IF (N .EQ. 9) GO TO 200                                           BRK10300
      CALL SIGMA(X,M,2)                                                 BRK10400
      STO1 = 1.414214*SIGZ                                              BRK10500
      TMPQ1 = 1.0/STO1                                                  BRK10600
      IF (SIGYNK) 200,200,147                                           BRK10700
  147 IF (SIGZ) 200,200,148                                             BRK10800
  148 LAT = Y/SIGYNK                                                    BRK10900
      LAT = -0.5*LAT*LAT                                                BRK11000
      IF (LAT .LT. -60.0) GO TO 200                                     BRK11100
      XBARX = EXP(LAT)                                                  BRK11200
      A = Z(M+1)-ZZL(K)                                                 BRK11300
      B = ZZL(K)-Z(M)                                                   BRK11400
      C = Z(M+1)+ZZL(K)-Z(IBOT)-Z(IBOT)                                 BRK11500
      D = Z(IBOT)+Z(IBOT)-Z(M)-ZZL(K)                                   BRK11600
      ERFX(1) = A*TMPQ1                                                 BRK11700
      ERFX(2) = B*TMPQ1                                                 BRK11800
      ERFX(3) = C*TMPQ1                                                 BRK11900
      ERFX(4) = D*TMPQ1                                                 BRK12000
      CALL ISO(1,4)                                                     BRK12100
      STO2 = ERFX(1)+ERFX(2)+GAMMA(1)*(ERFX(3)+ERFX(4))                 BRK12200
```

```
      S1 = 0.0                                                          BRK12300
      S3 = Z(ITOP+1)-Z(IBOT)                                            BRK12400
      E = 1.0                                                           BRK12500
      F = GAMMA(1)                                                      BRK12600
      G = F*F                                                           BRK12700
      IFL = 0                                                           BRK12800
  150 S1 = S1+2.0                                                       BRK12900
      S2 = S1*S3                                                        BRK13000
      ERFX(3) = (S2+D)*TMPQ1                                            BRK13100
      ERFX(4) = (C-S2)*TMPQ1                                            BRK13200
      IF (IFL .EQ. 0) GO TO 155                                         BRK13300
      IF (ERFX(3) .GT. 3.0 .AND. ERFX(4) .LT. -3.0) GO TO 185           BRK13400
  155 IFL = 1                                                           BRK13500
      ERFX(1) = (S2+B)*TMPQ1                                            BRK13600
      ERFX(2) = (A-S2)*TMPQ1                                            BRK13700
      CALL ISO(1,4)                                                     BRK13800
      STO2 = STO2+F*(ERFX(1)+ERFX(2))+E*(ERFX(3)+ERFX(4))               BRK13900
      ERFX(1) = (S2+A)*TMPQ1                                            BRK14000
      ERFX(2) = (B-S2)*TMPQ1                                            BRK14100
      ERFX(3) = (S2+C)*TMPQ1                                            BRK14200
      ERFX(4) = (D-S2)*TMPQ1                                            BRK14300
      CALL ISO(1,4)                                                     BRK14400
      STO2 = STO2+F*(ERFX(1)+ERFX(2))+G*(ERFX(3)+ERFX(4))               BRK14500
      E = F                                                             BRK14600
      F = G                                                             BRK14700
      G = G*GAMMA(1)                                                    BRK14800
      GO TO 150                                                         BRK14900
  185 CONTINUE                                                          BRK15000
      STO3 = 1.0/(Z(M+1)-Z(M))                                          BRK15100
      XBARX = EXP(-.5*(Y/SIGYNK)**2)                                    BRK15200
  190 TMPQ2 = X/UBAR(JF)                                                BRK15300
      S1 = (Q(M)*STO3 /(2.0*SQR2P*UBAR(JF)*SIGYNK))*XBARX*STO2          BRK15400
      IF (DECAY .GT. 0.0) S1 = S1*EXP(-DECAY*TMPQ2)                     BRK15500
      IF (LAMBDA .LE. 0.0.OR.TIM1.GE.TMPQ2+TAST(ILK-1)) GO TO 195       BRK15600
      IF (Z(M) .GT. ZLIM) GO TO 195                                     BRK15700
      S1 = S1*EXP(-LAMBDA*(TMPQ2+TAST(ILK-1)-TIM1))                     BRK15800
  195 CONTINUE                                                          BRK15900
      IF (SIGXNK) 210,210,211                                           BRK16000
  210 S2 = 0.0                                                          BRK16100
      GO TO 212                                                         BRK16200
  211 CONTINUE                                                          BRK16300
```

```
      S2 = (S1*UBAR(JF)/(SQR2P*SIGXNK))                                 BRK16400
  212 CONTINUE                                                          BRK16500
      DOS(J) = DOS(J)+S1                                                BRK16600
      CON(J) = CON(J)+S2                                                BRK16700
  200 CONTINUE                                                          BRK16800
      ANG(1) = UBAR(JF)                                                 BRK16900
      ANG(2) = SIGXNK                                                   BRK17000
  310 CONTINUE                                                          BRK17100
      AVCON(J) = 0.0                                                    BRK17200
      PASSTM(J) = 0.0                                                   BRK17300
      IF (ANG(2) .LE. 0.0) GO TO 311                                    BRK17400
      IF (DOS(J) .LE. 0.0) GO TO 311                                    BRK17500
      ERFX(1) = ANG(1)*TIMAV/(2.8284271*ANG(2))                         BRK17600
      CALL ISO(1,1)                                                     BRK17700
      AVCON(J) = (DOS(J)/TIMAV)*ERFX(1)                                 BRK17800
      PASSTM(J) = 4.3*ANG(2)/ANG(1)                                     BRK17900
  311 CONTINUE                                                          BRK18000
      RETURN                                                            BRK18100
      END                                                               BRK18200
```

```
      SUBROUTINE DEPOS                                                  DEP00100
      COMMON /PARAMT/ TESTNO(12),         ISKIP(15),NXS,NYS,NZS,NDI,NCI, DEP00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     DEP00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      DEP00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     DEP00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), DEP00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),DEP00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)DEP00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  DEP00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),DEP01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,     DEP01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, DEP01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    DEP01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          DEP01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       DEP01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   DEP01600
     7NYSS,CDAMX(3)                                                     DEP01700
C     **** THIS SUBROUTINE CALCULATES GRAVITATIONAL DEPOSITION AT GROUNDDEP01800
C                                                                       DEP01900
C                                                                       DEP02000
      DIMENSION DTHK(21)                                                DEP02100
      EQUIVALENCE (DTHK,XAST)                                           DEP02200
      REAL MPWR,L,LAMBDA                                                DEP02300
      INTEGER TESTNO                                                    DEP02400
C                                                                       DEP02500
      WRITE (6,905)                                                     DEP02600
      NBK = 0                                                           DEP02700
      DO 10 I=1,NNZ                                                     DEP02800
C                         LIST INPUT PARAMETERS                         DEP02900
      IF (I .GT. 1) GO TO 7                                             DEP03000
      WRITE (6,900) I,UBARK(I),UBARK(I+1),SIGAK(I),SIGAK(I+1),SIGEK(I), DEP03100
     1SIGEK(I+1),Q(I),DELX(I),DELY(I),SIGYO(I),SIGZO(I),ALPHA(I),BETA(I)DEP03200
     2,THETAK(I),TAUK,TAUOK,Z(I),THETAK(I+1)                            DEP03300
      GO TO 8                                                           DEP03400
    7 CONTINUE                                                          DEP03500
      WRITE (6,906) I,UBARK(I+1),SIGAK(I+1),SIGEK(I+1),Q(I),DELX(I),    DEP03600
     1DELY(I),SIGYO(I),SIGZO(I),ALPHA(I),BETA(I),Z(I),THETAK(I+1)       DEP03700
    8 IF (I .LT. NNZ) GO TO 9                                           DEP03800
      WRITE (6,907) Z(I+1)                                              DEP03900
    9 SIGAP(I) = SIGAK(I)*(TAUK/TAUOK)**(0.2)                           DEP04000
```

```
   10 CONTINUE                                                          DEP04100
      WRITE (6,901) (I,VS(I),I,PERC(I),I=1,NVS)                         DEP04200
      WRITE (6,902) (I,GAMMAP(I),I=1,NVS)                               DEP04300
      IF (NVB .LE. 0) GO TO 12                                          DEP04400
      WRITE (6,908) HB,(I,VB(I),I,PERCB(I),I=1,NVB)                     DEP04500
   12 CONTINUE                                                          DEP04600
      THETA(1) = THETAK(1)                                              DEP04700
      IF (THETA(1) .LT. 180.0) THET = (THETA(1)+180.0)*RAD              DEP04800
      IF (THETA(1) .GE. 180.0) THET = (THETA(1)-180.0)*RAD              DEP04900
      DTHK(1) = 0.0                                                     DEP05000
      DO 20 N=2,NZS                                                     DEP05100
   20 DTHK(N) = DTHK(N-1)+DELTHP(N-1)                                   DEP05200
      NYSS = NYS                                                        DEP05300
      IF (NYS .GT. 0) GO TO 23                                          DEP05400
      S = THETA(1)+0.5*DTHK(NZS)/FLOAT(NNZ)+180.0                       DEP05500
      S = AMOD(S,360.0)                                                 DEP05600
      S = AMOD(S,360.0)                                                 DEP05700
      NYS = 41                                                          DEP05800
      NYSS = 41                                                         DEP05900
      DO 22 J=1,NYS                                                     DEP06000
   22 YY(J) = YSV(J)+S                                                  DEP06100
   23 CONTINUE                                                          DEP06200
      DO 25 N=2,NZS                                                     DEP06300
   25 DTHK(N) = DTHK(N)*RAD                                             DEP06400
      DO 30 J=1,NYS                                                     DEP06500
      DO 30 I=1,NXS                                                     DEP06600
   30 DEPN(I,J) = 0.0                                                   DEP06700
      NTAD = 1                                                          DEP06800
      NTAL = 1                                                          DEP06900
      IF (NVB .GT. 0) NTAD = 2                                          DEP07000
      IF (NVB .LE. 0.AND.NNZ .EQ. 1) NTAL = 2                           DEP07100
      DO 73 JF=NTAL,NTAD                                                DEP07200
      NTAP = NVS                                                        DEP07300
      IF (JF .EQ. 2) NTAP = NVB                                         DEP07400
      DO 73 II=1,NTAP                                                   DEP07500
      IF (JF .EQ. 2.OR.VS(II) .LE. 10.0) GO TO 35                       DEP07600
      WRITE (6,903) VS(II)                                              DEP07700
      RETURN                                                            DEP07800
   35 CONTINUE                                                          DEP07900
      NTAK = 1                                                          DEP08000
      NTAR = NNZ                                                        DEP08100
```

```
      IF (NVB .LE. 0) GO TO 45                                    DEP08200
      IF (JF .EQ. 2) GO TO 40                                     DEP08300
      NTAR = NTAR-1                                               DEP08400
      GO TO 45                                                    DEP08500
   40 NTAK = NN2                                                  DEP08600
   45 DO 72 KK=NTAK,NTAR                                          DEP08700
      IF (JF .EQ. 2) GO TO 50                                     DEP08800
      IZ = 1                                                      DEP08900
      S = ((Z(KK+1)-Z(KK))*.3333333)+Z(KK)                        DEP09000
      CALL SGP(S,KK,SIGENK(1),1,IDMY,DMY,DMY,1)                   DEP09100
      CALL SGP(S,KK,DMY,2,IZ,UBHK,DMY,2)                          DEP09200
C     DETERMINE NO. SOURCES IN LINE SOURCE SIMULATION             DEP09300
      DHK = ACCUR*SIGENK(1)*SQBAR*SQRT(1.0+VS(II)/UBHK)           DEP09400
      IF (DHK .LT. 10.0) DHK = 10.0                               DEP09500
      S = (Z(KK+1)-Z(KK))/DHK                                     DEP09600
      NXCI = S+1.0                                                DEP09700
      IF (NXCI .LT. 3) NXCI = 3                                   DEP09800
      IF (NXCI .GT. 40) NXCI = 40                                 DEP09900
      IF (JF .EQ. 1) WRITE (6,909) VS(II),KK,NXCI                 DEP10000
      DHK = (Z(KK+1)-Z(KK))/FLOAT(NXCI)                           DEP10100
      ST01 = Z(KK)                                                DEP10200
      GO TO 55                                                    DEP10300
   50 NXCI = 1                                                    DEP10400
      ST01 = 0.0                                                  DEP10500
      DHK = HB                                                    DEP10600
   55 DO 60 IZ=1,NXCI                                             DEP10700
      ST01 = ST01+DHK                                             DEP10800
      ZZL(IZ) = ST01                                              DEP10900
      CALL SGP(ZZL(IZ),KK,SIGENK(IZ),1,IDMY,DMY,DMY,1)            DEP11000
      CALL SGP(ZZL(IZ),KK,SIGANK(IZ),2,IDMY,DMY,DMY,1)            DEP11100
      CALL SGP(ZZL(IZ),KK,DMY,2,IZ,UBHK,DMY,2)                    DEP11200
      CALL SGP(ZZL(IZ),KK,DMY,2,IZ,DMY,DMY,4)                     DEP11300
   60 CONTINUE                                                    DEP11400
      DO 71 I=1,NXS                                               DEP11500
      DO 71 J=1,NYS                                               DEP11600
      CALL COORD(N,1,X,Y,XX(I),YY(J),ASP,XS,1)                    DEP11700
      IF (N .EQ. 9) GO TO 71                                      DEP11800
      DO 70 IZ=1,NXCI                                             DEP11900
      PHI = ABS(ASP-(THET+ANG(IZ)))                               DEP12000
      IF (PHI .GT. 3.1415926536) PHI = 6.2831853072-PHI           DEP12100
      Y =XS*SIN(PHI)                                              DEP12200
```

```
      X =XS*COS(PHI)                                                    DEP12300
      IF (X .LE. 0.0) GO TO 70                                          DEP12400
      DEP = 0.0                                                         DEP12500
      CALL SGP(DMY,KK,DMY,2,IZ,DMY,X,3)                                 DEP12600
      IF (SIGYNK .LE. 0.0) GO TO 70                                     DEP12700
      DMY = -0.5*(Y/SIGYNK)**2                                          DEP12800
      IF (DMY .LT. -30.0) GO TO 70                                      DEP12900
      DEP = DEP*EXP(DMY)                                                DEP13000
      DEPN(I,J) = DEPN(I,J)+DEP                                         DEP13100
   70 CONTINUE                                                          DEP13200
   71 CONTINUE                                                          DEP13300
   72 CONTINUE                                                          DEP13400
   73 CONTINUE                                                          DEP13500
C     OUTPUT GRAVITATIONAL DEPOSITION                                   DEP13600
      DO 80 I=1,NXS                                                     DEP13700
      DO 80 J=1,NYSS                                                    DEP13800
   80 CDAMX(1) = AMAX1(CDAMX(1),DEPN(I,J))                              DEP13900
      MDLS = 6                                                          DEP14000
      ZZL(1) = Z(1)                                                     DEP14100
      CALL GENPRT(1)                                                    DEP14200
C                                                                       DEP14300
      DO 90 I=1,NXS                                                     DEP14400
      DO 90 J=1,NYS                                                     DEP14500
      DEPN(I,J) = 0.0                                                   DEP14600
   90 CONTINUE                                                          DEP14700
      RETURN                                                            DEP14800
  900 FORMAT (23H0*** DATA INPUTS LAYER ,I2,18H, UBARK AT BOTTOM=,F8.4,1DEP14900
     15H, UBARK AT TOP=,F8.4,18H, SIGAK AT BOTTOM=,F8.5,15H, SIGAK AT TODEP15000
     2P=,F8.5/17H SIGEK AT BOTTOM=,F8.5,15H, SIGEK AT TOP=,F8.5,4H, Q=,EDEP15100
     314.8,7H, DELX=,E14.8,7H, DELY=,E14.8/7H SIGYO=,F9.4,8H, SIGZO=,F9.DEP15200
     44,6H, ALPHA=,F4.1,7H, BETA=,F4.1,19H, THETAK AT BOTTOM=,F8.3,7H, TDEP15300
     5AUK=,F8.3,8H, TAUOK=,F8.3/4H  Z=,F9.3                             DEP15400
     6,16H, THETAK AT TOP=,F8.3)                                        DEP15500
  901 FORMAT (1H0,3(3HVS(,I2,2H)=,F10.5,7H, PERC(,I2,2H)=,F10.5,2H, )/(1DEP15600
     1X,3HVS(,I2,2H)=,F10.5,7H, PERC(,I2,2H)=,F10.5,5H, VS(,I2,2H)=,F10.DEP15700
     25,7H, PERC(,I2,2H)=,F10.5,5H, VS(,I2,2H)=,F10.5,7H, PERC(,I2,2H)=,DEP15800
     3F10.5,2H, ))                                                      DEP15900
  902 FORMAT ((1X,7(7HGAMMAP(,I2,2H)=,F5.3,2H, )))                      DEP16000
  903 FORMAT (1H0,67H****** ERROR ****** VS HAS EXCEEDED MAXIMUM ALLOWABDEP16100
     1LE VALUE 10, VS=,F9.4)                                            DEP16200
  905 FORMAT (1H1,48X,36H***** GRAVITATIONAL DEPOSITION *****)          DEP16300
```

```
  906 FORMAT (8H0 LAYER ,I2,15H, UBARK AT TOP=,F8.4,15H, SIGAK AT TOP=,FDEP16400
     18.5,15H, SIGEK AT TOP=,F8.5,4H, Q=,E14.8,7H, DELX=,E14.8/6H DELY=,DEP16500
     2E14.8,8H, SIGYO=,F9.4,8H, SIGZO=,F9.4,8H, ALPHA=,F4.1,7H, BETA=,F4DEP16600
     3.1,4H, Z=,F9.3/15H THETAK AT TOP=,F8.3)                          DEP16700
  907 FORMAT (1X,10H Z AT TOP=,F10.4)                                  DEP16800
  908 FORMAT (1X,19HHEIGHT OF BURST HB=,F10.4,3HVB(,I2,2H)=,F10.5,8H, PEDEP16900
     1RCB(,I2,2H)=,F10.5,2H, /(1X,3HVB(,I2,2H)=,F10.5,8H, PERCB(,I2,2H)=DEP17000
     2,F10.5,5H, VB(,I2,2H)=,F10.5,8H, PERCB(,I2,2H)=,F10.5,5H, VB(,I2, DEP17100
     32H)=,F10.5,8H, PERCB(,I2,2H)=,F10.5,2H, ))                       DEP17200
  909 FORMAT (1H0,10X,4HVS =,F8.4,12H, LAYER NO. ,I2,18H, NO. OF SOURCESDEP17300
     1 =,I6)                                                           DEP17400
      END                                                              DEP17500
```

```
      SUBROUTINE WASHT                                                  WSH00100
      COMMON /PARAMT/ TESTNO(12),         ISKIP(15),NXS,NYS,NZS,NDI,NCI, WSH00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     WSH00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      WSH00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     WSH00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), WSH00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),WSH00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)WSH00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  WSH00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),WSH01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    WSH01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, WSH01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    WSH01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          WSH01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       WSH01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   WSH01600
     7NYSS,CDAMX(3)                                                     WSH01700
      DIMENSION WASHOU(41,1)                                            WSH01800
      EQUIVALENCE (DEPN,WASHOU)                                         WSH01900
      EQUIVALENCE (ISW6,SIGENK),(A,SIGENK(2)),(B,SIGENK(3)),(C,SIGENK(4)WSH02000
     1),(D,SIGENK(5)),(E,SIGENK(6)),(G,SIGENK(7))                       WSH02100
      REAL MPWR,L,LAMBDA                                                WSH02200
      INTEGER TESTNO                                                    WSH02300
C     THIS SUBROUTINE CALCULATES PRECIPITATION DEPOSITION - MODEL 5     WSH02400
      C = 1.0                                                           WSH02500
      D = 1.0                                                           WSH02600
      E = 1.0                                                           WSH02700
      CALL COORD(N,KK,X,Y,XX(I),YY(J),ASP,XS,1)                         WSH02800
      IF (NBK.NE.0.AND.IBOT.LE.KK.AND.KK.LE.ITOP) GO TO 20              WSH02900
      IF (N .EQ. 9) GO TO 70                                            WSH03000
   10 CALL SIGMA(X,KK,1)                                                WSH03100
      A = UBAR(KK)                                                      WSH03200
      B = SIGY                                                          WSH03300
      G = TIM1                                                          WSH03400
      GO TO 30                                                          WSH03500
   20 IF (N .NE. 9) GO TO 10                                            WSH03600
      CALL COORD(N,KK,X,Y,XX(I),YY(J),ASP,XS,2)                         WSH03700
      IF (N .EQ. 9) GO TO 70                                            WSH03800
      CALL SIGMA(X,KK,2)                                                WSH03900
      A = UBAR(JF)                                                      WSH04000
```

```
      B = SIGYNK                                                          WSH04100
      G = TIM1-TAST(ILK-1)                                                WSH04200
      SIGX = SIGXNK                                                       WSH04300
   30 IF (X/A .LT. G) GO TO 70                                            WSH04400
      IF (B .LE. 0.0) GO TO 70                                            WSH04500
      IF (G .LT. (X-2.15*SIGX)/A) GO TO 40                                WSH04600
      IF (ISKIP(4) .EQ. 1) GO TO 40                                       WSH04700
       E = AMOD(YY(J),360.0)                                              WSH04800
      IF (E .LT. 0.0) E = 360.0+E                                         WSH04900
      WRITE (6,80) XX(I),E                                                WSH05000
   40 IF (MODLS(KK) .EQ. 3) GO TO 50                                      WSH05100
   50 E = Y/B                                                             WSH05200
      E = -0.5*E*E                                                        WSH05300
      IF (E .LT. -60.0) GO TO 70                                          WSH05400
      E = EXP(E)                                                          WSH05500
      IF (ISKIP(4) .EQ. 1) GO TO 60                                       WSH05600
      C = EXP(-LAMBDA*(X/A-G))                                            WSH05700
   60 WASHOU(I,J) = WASHOU(I,J)+(LAMBDA*Q(KK)/(SQR2P*A*B))*C*E            WSH05800
   70 RETURN                                                              WSH05900
   80 FORMAT (1H0,36H *** PRECIPITATION DEPOSITION AT XX=,F10.3,5H, YY=,WSH06000
     1F10.3,26H MAY BE OVER ESTIMATED ***/)                               WSH06100
      END                                                                 WSH06200
```

```
      SUBROUTINE GENPRT(K)                                              GPT00100
      COMMON /PARAMT/ TESTNO(12),          ISKIP(15),NXS,NYS,NZS,NDI,NCI, GPT00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     GPT00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      GPT00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     GPT00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), GPT00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),GPT00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)GPT00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  GPT00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),GPT01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    GPT01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, GPT01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    GPT01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          GPT01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       GPT01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   GPT01600
     7NYSS,CDAMX(3)                                                     GPT01700
C     THIS PROGRAM CONTROLS PRINTING OF ALL PROGRAM CALCULATIONS        GPT01800
      DIMENSION LINE(1),YB(1),DPN(41,1),IX(1)                           GPT01900
      DIMENSION JLINE(70),KLINE(10)                                     GPT02000
      REAL LAMBDA                                                       GPT02100
      COMMON /LBLLBL/ J1(9),J2(4),J3(48),J5(6),J7(3),J8(16),J9(13),J10, GPT02200
     1J4(12),J11(2),UNIT(15)                                            GPT02300
      EQUIVALENCE (YBARY,LINE),(YB,SIGENK),(DPN,BETANK),(IX,XI)         GPT02400
      COMMON /BNDS/ XRIT,XLFT,YBOT,YTOP,XPL,YPL                         GPT02500
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, GPT02600
     1XSIZE1,YSIZE1                                                     GPT02700
      COMMON /PLTLLO/ ISW,XMAXJN,YMAXJN,XCIZE,YCIZE                     GPT02800
      COMMON /XYXYPT/ YP(41),XP(41),A(41),B(41),C(41),D(41),XI(41),YI(41GPT02900
     1),NUM(3),NC                                                       GPT03000
      DATA ISP/1H0/,JSP/1H /,MS/57/                                     GPT03100
      DATA UNIT/1H ,5H(PPM),1H ,6H    (P,6HPM SEC,1H),6H    (S,6HECONDS,GPT03200
     11H),6H    (M,6HG/M**3,1H),5H  (MG,6HSEC/M*,3H*3)/                 GPT03300
      JM = 5                                                            GPT03400
      IT1 = 1                                                           GPT03500
      IT2 = 7                                                           GPT03600
      IT3 = 13                                                          GPT03700
      NCV1 = 14                                                         GPT03800
      NCV2 = 7                                                          GPT03900
      NCV3 = 24                                                         GPT04000
```

```
      IF (KSW(1) .LE. 0) GO TO 10                                       GPT04100
      IT1 = 19                                                          GPT04200
      GO TO 20                                                          GPT04300
   10 IF (KSW(2) .LE. 0) GO TO 30                                       GPT04400
      IT1 = 31                                                          GPT04500
   20 NCV1 = 25                                                         GPT04600
      NCV2 = -1                                                         GPT04700
      NCV3 = -1                                                         GPT04800
      JM = 1                                                            GPT04900
   30 CONTINUE                                                          GPT05000
      XPRT = TIMAV/60.0                                                 GPT05100
      NXSS = NXS-2                                                      GPT05200
      IF (ISKIP(1) .LE. 0) GO TO 170                                    GPT05300
C     PRINT GENERAL GRID CALCULATIONS                                   GPT05400
C     GET Y IN PROPER INTERVAL                                          GPT05500
      DO 100 J=1,NYSS                                                   GPT05600
      YB(J) = AMOD(YY(J),360.0)                                         GPT05700
      IF (YB(J) .LT. 0.0) YB(J) = 360.0+YB(J)                           GPT05800
  100 CONTINUE                                                          GPT05900
      IB = 0                                                            GPT06000
      DO 160 KS=1,JM                                                    GPT06100
      IF (JM .EQ. 1) GO TO 110                                          GPT06200
      IF (KS .EQ. 5) IB = 1                                             GPT06300
      CALL INPTS(KS,IB,NXS,II,NYSS,DEPN,SIGANK)                         GPT06400
  110 CALL HEDING(KSW,KS,1,0)                                           GPT06500
      CALL LABELS(K)                                                    GPT06600
      CALL VRTCLE(KS,JM,KSW,SIGANK,ISKIP(5),NCV)                        GPT06700
      N1 = -9                                                           GPT06800
  120 N1 = N1+10                                                        GPT06900
      N2 = N1+9                                                         GPT07000
      IF (N1 .GT. NYSS) GO TO 160                                       GPT07100
      IF (N2 .GT. NYSS) N2 = NYSS                                       GPT07200
      LINES = 80                                                        GPT07300
      DO 150 I=1,NXSS                                                   GPT07400
      LINES = LINES+1                                                   GPT07500
      IF (LINES .LT. MS) GO TO 140                                      GPT07600
      IF (JM .GT. 1) GO TO 125                                          GPT07700
      CALL PRTTTL(NWD,LINES,LINE,0.0,0.0)                               GPT07800
      GO TO 126                                                         GPT07900
  125 CALL PRTTTL(NWD,LINES,LINE,DECAY,LAMBDA)                          GPT08000
  126 CONTINUE                                                          GPT08100
```

```
      IF (KS .GT. 3) GO TO 130                                         GPT08200
      WRITE (6,900) CDAMX(KS),(SIGANK(J),J=1,NCV)                      GPT08300
      LINES = LINES+2                                                  GPT08400
  130 WRITE (6,901) (YB(J),J=N1,N2)                                    GPT08500
      WRITE (6,902) (SIGANK(J),J=1,NCV)                                GPT08600
      LINES = LINES+4                                                  GPT08700
  140 WRITE (6,903) XX(I),(DEPN(I,J),J=N1,N2)                          GPT08800
  150 CONTINUE                                                         GPT08900
      IF (N2 .LT. NYSS) GO TO 120                                      GPT09000
  160 CONTINUE                                                         GPT09100
  170 CONTINUE                                                         GPT09200
      IF (JM .GT. 1) GO TO 190                                         GPT09300
      DO 180 I=1,NXS                                                   GPT09400
      KOUT = 4*I-3                                                     GPT09500
      CALL INTOUT(DEPN,KOUT,NYSS,2,41,I)                               GPT09600
  180 CONTINUE                                                         GPT09700
  190 CONTINUE                                                         GPT09800
C     PRINT AND/OR PLOT CENTERLINE CALCULATIONS                        GPT09900
      IF (ISKIP(2) .LE. 0) GO TO 480                                   GPT10000
      IB = 0                                                           GPT10100
      DO 340 KS=1,JM                                                   GPT10200
      IF (JM .EQ. 1) GO TO 250                                         GPT10300
      IF (KS .EQ. 5) IB = 1                                            GPT10400
      CALL INPTS(KS,IB,NXS,II,NYSS,DEPN,SIGANK)                        GPT10500
  250 CONTINUE                                                         GPT10600
      DO 340 I=1,NXS                                                   GPT10700
      I1 = IX(I)                                                       GPT10800
      IF (KS .GT. 1) GO TO 270                                         GPT10900
      IX(I) = NYSS/2+1                                                 GPT11000
      YMAX = 0.0                                                       GPT11100
C     FIND INDEX AT OR CLOSE TO MAXIMUM                                GPT11200
      DO 260 J=1,NYSS                                                  GPT11300
      IF (DEPN(I,J) .LE. YMAX) GO TO 260                               GPT11400
      IX(I) = J                                                        GPT11500
      YMAX = DEPN(I,J)                                                 GPT11600
  260 CONTINUE                                                         GPT11700
      I1 = IX(I)                                                       GPT11800
      YP(I) = YY(I1)                                                   GPT11900
  270 I2 = MAX0(1,I1-3)                                                GPT12000
      I3 = MIN0(NYSS,I1+3)                                             GPT12100
      I3 = I3-I2+1                                                     GPT12200
```

```
      DPN(I,KS) = DEPN(I,I1)                                            GPT12300
      IF (I3 .LT. 3) GO TO 340                                          GPT12400
      DO 280 J=1,I3                                                     GPT12500
  280 XP(J) = DEPN(I,I2+J-1)                                            GPT12600
      CALL SPLINE(YY(I2),XP,A,B,C,D,I3,IER)                             GPT12700
      IF (IER .EQ. 1) GO TO 340                                         GPT12800
      IF (KS .GT. 1) GO TO 310                                          GPT12900
      J = 1                                                             GPT13000
      YMAX = 0.0                                                        GPT13100
      YPL = YY(I2)-0.1                                                  GPT13200
  290 YPL = YPL+0.1                                                     GPT13300
      IF (YPL .LT. YY(I2+J)) GO TO 300                                  GPT13400
      J = J+1                                                           GPT13500
  300 IF (J .GE. I3) GO TO 340                                          GPT13600
      XPL = YPL-YY(I2+J-1)                                              GPT13700
      XPL = XP(J)+XPL*(B(J)+(YPL-YY(I2+J))*(2.0*C(J)+C(J+1)+A(J)*XPL)*  GPT13800
     1.16666666)                                                        GPT13900
      IF (XPL .LE. YMAX) GO TO 290                                      GPT14000
      YMAX = XPL                                                        GPT14100
      YP(I) = YPL                                                       GPT14200
      DPN(I,KS) = YMAX                                                  GPT14300
      GO TO 290                                                         GPT14400
  310 DO 320 J=1,I3                                                     GPT14500
      IF (YP(I) .LT. YY(I2+J-1)) GO TO 330                              GPT14600
  320 CONTINUE                                                          GPT14700
  330 XPL = YP(I)-YY(I2+J-2)                                            GPT14800
      DPN(I,KS) = XP(J-1)+XPL*(B(J-1)+(YP(I)-YY(I2+J-1))*(2.0*C(J-1)+C(JGPT14900
     1)+A(J-1)*XPL)*.16666666)                                          GPT15000
  340 CONTINUE                                                          GPT15100
C     PRINT MAXIMUM CENTERLINE CALCULATIONS                             GPT15200
      IF (ISKIP(2) .EQ. 2) GO TO 420                                    GPT15300
      CALL HEDING(KSW,1,2,1)                                            GPT15400
      CALL LABELS(K)                                                    GPT15500
      LINES = 80                                                        GPT15600
      CDAMX(1) = 0.0                                                    GPT15700
      CDAMX(2) = 0.0                                                    GPT15800
      CDAMX(3) = 0.0                                                    GPT15900
      DO 350 I=1,NXSS                                                   GPT16000
      DO 350 J=1,3                                                      GPT16100
      CDAMX(J) = AMAX1(CDAMX(J),DPN(I,J))                               GPT16200
  350 CONTINUE                                                          GPT16300
```

```
      IF (JM .GT. 1) GO TO 370                                          GPT16400
      IF (KSW(1) .GT. 0) GO TO 360                                      GPT16500
      I1 = 31                                                           GPT16600
      I2 = 35                                                           GPT16700
      I3 = 33                                                           GPT16800
      GO TO 380                                                         GPT16900
  360 I1 = 19                                                           GPT17000
      I2 = 23                                                           GPT17100
      I3 = 21                                                           GPT17200
      GO TO 380                                                         GPT17300
  370 CONTINUE                                                          GPT17400
      I1 = 1                                                            GPT17500
      I2 = 6                                                            GPT17600
      I3 = 3                                                            GPT17700
      IF (ISKIP(5) .NE. 4) GO TO 380                                    GPT17800
      I1 = 10                                                           GPT17900
      I2 = 15                                                           GPT18000
      I3 = 12                                                           GPT18100
  380 CONTINUE                                                          GPT18200
      DO 410 I=1,NXSS                                                   GPT18300
      IF (DPN(I,1) .LE. 0.0) GO TO 410                                  GPT18400
      LINES = LINES+1                                                   GPT18500
      IF (LINES .LT. MS) GO TO 400                                      GPT18600
      IF (JM .GT. 1) GO TO 390                                          GPT18700
      CALL PRTTTL(NWD,LINES,LINE,0.0,0.0)                               GPT18800
      WRITE (6,904) ISP,CDAMX(1),(J3(J),J=I1,I2)                        GPT18900
      WRITE (6,905) (J3(J),J=I1,I3)                                     GPT19000
      LINES = LINES+6                                                   GPT19100
      GO TO 400                                                         GPT19200
  390 CALL PRTTTL(NWD,LINES,LINE,DECAY,LAMBDA)                          GPT19300
      WRITE (6,904) ISP,CDAMX(1),(J3(J),J=1,3)                          GPT19400
      WRITE (6,907) JSP,CDAMX(2),(J3(J),J=7,8)                          GPT19500
      WRITE (6,908) JSP,CDAMX(3),XPRT,(J3(J),J=13,16)                   GPT19600
      WRITE (6,906) XPRT,(UNIT(J),J=I1,I2),(UNIT(J),J=I1,I3),(UNIT(J),J=GPT19700
     17,9),(UNIT(J),J=I1,I3)                                            GPT19800
      LINES = LINES+8                                                   GPT19900
  400 YPL = AMOD(YP(I),360.0)                                           GPT20000
      IF (YPL .LT. 0.0) YPL = YPL+360.0                                 GPT20100
      WRITE (6,909) XX(I),YPL,(DPN(I,J),J=1,JM)                         GPT20200
  410 CONTINUE                                                          GPT20300
  420 CONTINUE                                                          GPT20400
```

```
      C     PLOT MAXIMUM CENTERLINE CALCULATIONS                          GPT20500
            IF (ISKIP(2) .EQ. 1) GO TO 480                                GPT20600
            HT = 0.0                                                      GPT20700
            XPL = 0.0                                                     GPT20800
            YPL = 0.0                                                     GPT20900
            DO 430 I=1,NXS                                                GPT21000
            X1 = XX(I)*SIN(YP(I)*RAD)                                     GPT21100
            Y1 = XX(I)*COS(YP(I)*RAD)                                     GPT21200
            YB(I) = SQRT((X1-XPL)**2+(Y1-YPL)**2)+HT                      GPT21300
            HT = YB(I)                                                    GPT21400
            XPL = X1                                                      GPT21500
        430 YPL = Y1                                                      GPT21600
            IF (ISW .NE. 2) GO TO 436                                     GPT21700
            DO 435 I=1,NXS                                                GPT21800
            YB(I) = ALOG10(YB(I))                                         GPT21900
        435 CONTINUE                                                      GPT22000
        436 CONTINUE                                                      GPT22100
            CALL FSTPLT(H,RNG,AZM,NAMCAS,IDATE,ITIME,CDAMX(1),CDAMX(2),CDAMX(3GPT22200
           1),J3(IT1),J3(IT2),J3(IT3),NCV1,NCV2,NCV3,XPRT)                GPT22300
            IF (JM .GT. 3) JM = 3                                         GPT22400
            DO 470 KS=1,JM                                                GPT22500
            CALL HEDING(KSW,KS,2,0)                                       GPT22600
            CALL LABELS(K)                                                GPT22700
            CALL VRTCLE(KS,JM,KSW,SIGANK,ISKIP(5),NCV)                    GPT22800
            NCV = NCV*6                                                   GPT22900
            GO TO (440,450,460),KS                                        GPT23000
        440 IF (KSW(1) .GT. 0.OR.KSW(2) .GT. 0) GO TO 450                 GPT23100
      C     PLOT MAXIMUM CENTERLINE CONCENTRATION                         GPT23200
            IF (NCCC .LE. 0) GO TO 470                                    GPT23300
            CALL LLPLOT(DPN(1,KS),YB,NXSS,LINE,CI,NCCC,NWD,SIGANK,NCV)    GPT23400
            NWD = NWD*6                                                   GPT23500
            CALL LSSOPT(YB,DPN(1,KS),NXSS,NCCC,CI,SIGANK,LINE,NCV,NWD)    GPT23600
            GO TO 470                                                     GPT23700
      C     CENTERLINE DEPOSITION OR DOSAGE                               GPT23800
        450 IF (NDDD .LE. 0) GO TO 470                                    GPT23900
            CALL LLPLOT(DPN(1,KS),YB,NXSS,LINE,DI,NDDD,NWD,SIGANK,NCV)    GPT24000
            NWD = NWD*6                                                   GPT24100
            CALL LSSOPT(YB,DPN(1,KS),NXSS,NDDD,DI,SIGANK,LINE,NCV,NWD)    GPT24200
            GO TO 470                                                     GPT24300
      C     CENTERLINE TIME-MEAN CONCENTRATION                            GPT24400
        460 IF (NTTT .LE. 0) GO TO 470                                    GPT24500
```

```
      CALL LLPLOT(DPN(1,KS),YB,NXSS,LINE,TI,NTTT,NWD,SIGANK,NCV)        GPT24600
      NWD = NWD*6                                                       GPT24700
      CALL LSSOPT(YB,DPN(1,KS),NXSS,NTTT,TI,SIGANK,LINE,NCV,NWD)        GPT24800
  470 CONTINUE                                                          GPT24900
  480 CONTINUE                                                          GPT25000
C     PLOT ISOPLETHS                                                    GPT25100
      IF (ISKIP(3) .LE. 0) GO TO 540                                    GPT25200
      IF (ISKIP(3) .EQ. 1) GO TO 485                                    GPT25300
      CALL FSTPLT(H,RNG,AZM,NAMCAS,IDATE,ITIME,CDAMX(1),CDAMX(2),CDAMX(3GPT25400
     1),J3(IT1),J3(IT2),J3(IT3),NCV1,NCV2,NCV3,XPRT)                    GPT25500
  485 CONTINUE                                                          GPT25600
      IF (JM .GT. 3) JM = 3                                             GPT25700
      DO 530 KS=1,JM                                                    GPT25800
      IF (JM .EQ. 1) GO TO 490                                          GPT25900
      CALL INPTS(KS,0,NXS,II,NYSS,DEPN,SIGANK)                          GPT26000
  490 CONTINUE                                                          GPT26100
      CALL HEDING(KSW,KS,3,0)                                           GPT26200
      CALL LABELS(K)                                                    GPT26300
      CALL VRTCLE(KS,JM,KSW,KLINE,ISKIP(5),NCV)                         GPT26400
      DO 495 J=1,NWD                                                    GPT26500
  495 JLINE(J) = LINE(J)                                                GPT26600
      NWD = NWD*6                                                       GPT26700
      GO TO (500,510,520),KS                                            GPT26800
  500 IF (KSW(1) .GT. 0.OR.KSW(2) .GT. 0) GO TO 510                     GPT26900
      IF (NCC .LE. 0) GO TO 530                                         GPT27000
      CALL ISSOPT(XX,YY,NXS,NCC,CI,JLINE,NWD,DEPN,NYSS,YBARY,DEPN,II,KS,GPT27100
     1YT,ISKIP(3),KLINE,NCV,JM,DECAY,LAMBDA)                            GPT27200
      GO TO 530                                                         GPT27300
  510 IF (NDD .LE. 0) GO TO 530                                         GPT27400
      CALL ISSOPT(XX,YY,NXS,NDD,DI,JLINE,NWD,DEPN,NYSS,YBARY,DEPN,II,KS,GPT27500
     1YT,ISKIP(3),KLINE,NCV,JM,DECAY,LAMBDA)                            GPT27600
      GO TO 530                                                         GPT27700
  520 IF (NTT .LE. 0) GO TO 530                                         GPT27800
      CALL ISSOPT(XX,YY,NXS,NTT,TI,JLINE,NWD,DEPN,NYSS,YBARY,DEPN,II,KS,GPT27900
     1YT,ISKIP(3),KLINE,NCV,JM,DECAY,LAMBDA)                            GPT28000
  530 CONTINUE                                                          GPT28100
  540 CONTINUE                                                          GPT28200
      RETURN                                                            GPT28300
  900 FORMAT (1H0,38X,F9.3,21H IS THE MAXIMUM GRID ,5A6)                GPT28400
  901 FORMAT (1H0,6H RANGE,44X,29H- AZIMUTH BEARING (DEGREES) -/1X,8H(MEGPT28500
     ITERS),10(3X,F7.2,2X))                                             GPT28600
```

```
  902 FORMAT (52X,6A6)                                                        GPT28700
  903 FORMAT (1X,F8.1,10(F10.3,2X))                                           GPT28800
  904 FORMAT (A1,38X,F9.3,20H IS THE MAXIMUM PEAK,5A6)                        GPT28900
  905 FORMAT (1H0,23X,12HMAXIMUM PEAK/18H  RANGE    AZIMUTH,6X,2A6,A2/11GPT29000
     1X,24HBEARING        DEPOSITION/1X,34H(METERS) (DEGREES)       (MG/MGPT29100
     2**2))                                                                   GPT29200
  906 FORMAT (1H0,23X,12HMAXIMUM PEAK,13X,7HMAXIMUM,13X,12HMAXIMUM PEAK/GPT29300
     17H  RANGE,4X,7HAZIMUTH,6X,13HCONCENTRATION,13X,6HDOSAGE,8X,F5.1,17GPT29400
     2H MINUTE TIME-MEAN,8X,7HTIME OF,12X,13HAVERAGE CLOUD/11X,7HBEARINGGPT29500
     3,51X,13HCONCENTRATION,9X,13HCLOUD PASSAGE,9X,13HCONCENTRATION/1X,1GPT29600
     49H(METERS) (DEGREES) ,5(2X,3A6,2X))                                     GPT29700
  907 FORMAT (A1,44X,F9.3,15H IS THE MAXIMUM,5A6)                             GPT29800
  908 FORMAT (A1,32X,F9.3,20H IS THE MAXIMUM PEAK,F5.1,7H MINUTE,5A6)   GPT29900
  909 FORMAT (1X,F8.1,2X,F6.1,8X,F10.3,11X,F10.3,13X,F10.3,2(12X,F10.3))GPT30000
      END                                                                     GPT30100
```

```
      SUBROUTINE READER(IFF)                                            RDR00100
      COMMON /PARAMT/ TESTNO(12),       ISKIP(15),NXS,NYS,NZS,NDI,NCI, RDR00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     RDR00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      RDR00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     RDR00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), RDR00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VR(20),PERCR(20),RDR00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)RDR00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  RDR00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),RDR01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    RDR01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, RDR01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    RDR01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          RDR01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       RDR01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   RDR01600
     7NYSS,CDAMX(3)                                                     RDR01700
C     THIS SUBROUTINE READS ALL INPUT DATA AND CALCULATES NECESSARY     RDR01800
C     LAYER PARAMETERS                                                  RDR01900
      INTEGER TESTNO                                                    RDR02000
      REAL MPWR,L,LAMBDA                                                RDR02100
      DIMENSION XSV(41),IZR1(1)                                         RDR02200
      DIMENSION TEMPK(16),TEMPL(10)                                     RDR02300
      DIMENSION NTFB(2)                                                 RDR02400
      COMMON /PLTISO/ SCL,XMAXIN,YMAXIN,XSIZE,YSIZE,RASTIN,JSW          RDR02500
      COMMON /PLTLLO/ ISW,XMAXJN,YMAXJN,XCIZE,YCIZE                     RDR02600
      EQUIVALENCE(NTFB,ITOP)                                            RDR02700
      EQUIVALENCE (I1,CON),(I2,CON(2)),(I3,CON(3)),(NTAL,CON(4)),(NTAK, RDR02800
     1CON(5)),(INNZ,CON(6)),(S,CON(7)),(S1,CON(8)),(P,CON(9)),(M,CON(10)RDR02900
     2),(NNZI,CON(11)),(N,CON(12)),(DIF1,CON(13)),(DIF2,CON(14))        RDR03000
      EQUIVALENCE (IZR1,ISKIP)                                          RDR03100
      DATA YSV/-40.,-35.,-30.,-27.,-24.,-22.,-20.,-18.,-16.,-14.,-12.,  RDR03200
     1-10.,-8.,-7.,-6.,-5.,-4.,-3.,-2.,-1.,0.,1.,2.,3.,4.,5.,6.,7.,8.,  RDR03300
     210.,12.,14.,16.,18.,20.,22.,24.,27.,30.,35.,40./                  RDR03400
      DATA XSV/500.,1250.,2500.,3750.,5000.,6250.,7500.,8750.,10000.,   RDR03500
     111250.,12500.,13750.,15000.,16250.,17500.,18750.,20000.,21250.,   RDR03600
     222500.,23750.,25000.,26250.,27500.,28750.,30000.,31250.,32500.,   RDR03700
     333750.,35000.,36250.,37500.,38750.,40000.,41250.,42500.,43750.,   RDR03800
     445000.,47500.,50000.,65000.,80000./                               RDR03900
C     MACHINE DEPENDENT STATEMENT ASSUMES SIX BYTES/WORD                RDR04000
```

```
      DATA TESTNO/12*6H      /,NAMCAS/12*6H      /                      RDR04100
C     SR12I =  1.0/SQRT(12.0)                                           RDR04200
      NAMELIST /NAM2/ TESTNO,ISKIP,NXS,NYS,NZS,NDI,NCI,NPTS,NTI,TI,     RDR04300
     1NVS,NVB,XX,YY,Z,DELX,DELY,Q,UBARK,SIGAK,SIGEK,SIGXO,SIGYO,GAMMAP, RDR04400
     2SIGZO,ALPHA,BETA,ZRK,TIMAV,THETAK,TAUK,TAUOK,H,XRY,XRZ,XLRY,XLRZ, RDR04500
     3ZZL,IZMOD,DECAY,TIM1,BLAMDA,DI,CI,TAST,ZLIM,HB,PERCB,VB,          RDR04600
     4VS,PERC,ACCUR,ALPHL,BETL,TAUL,TAUOL,ZRL,UBARL,SIGAL,SIGEL,THETAL, RDR04700
     5NPS,NAMCAS,SCL,XMAXIN,YMAXIN,ISW,XMAXJN,YMAXJN,RASTIN             RDR04800
     6,XSIZE,YSIZE,XCIZE,YCIZE,TEMPK,TEMPL,JSW                          RDR04900
      IF (IFF .GT. 1) GO TO 2                                           RDR05000
C     ZERO OUT INPUT LISTS FOR PROCESSORS WHERE CORE IS NOT             RDR05100
C     INITIALIZE TO ZERO, 608 IS THE LENGTH OF COMMON /PARAMT/, SUBTRACTRDR05200
C     12 FOR TESTNO AND 12 FOR NAMCAS                                   RDR05300
      DO 1 I=1,584                                                      RDR05400
    1 IZR1(I) = 0                                                       RDR05500
    2 READ (5,NAM2)                                                     RDR05600
      DO 71 I=1,20                                                      RDR05700
   71 GAMMA(I) = 1.0-GAMMAP(I)                                          RDR05800
      NNZ = NZS-1                                                       RDR05900
      LAMBDA = BLAMDA                                                   RDR06000
      NCC = NCI/10                                                      RDR06100
      NDD = NDI/10                                                      RDR06200
      NTT = NTI/10                                                      RDR06300
      NCCC = NCI-NCC*10                                                 RDR06400
      NDDD = NDI-NDD*10                                                 RDR06500
      NTTT = NTI-NTT*10                                                 RDR06600
      IF (ISW .LE. 0) ISW = 2                                           RDR06700
      IF (RASTIN .LE. 0.0) RASTIN = 163.2                               RDR06800
      IF (XSIZE .LE. 0.0) XSIZE = 937.0                                 RDR06900
      IF (YSIZE .LE. 0.0) YSIZE = 899.0                                 RDR07000
      IF (XCIZE .LE. 0.0) XCIZE = 937.0                                 RDR07100
      IF (YCIZE .LE. 0.0) YCIZE = 899.0                                 RDR07200
      IF (NXS .GT. 0) GO TO 5                                           RDR07300
C     DEFAULT XX                                                        RDR07400
      NXS = 41                                                          RDR07500
      DO 4 I=1,NXS                                                      RDR07600
    4 XX(I) = XSV(I)                                                    RDR07700
    5 CONTINUE                                                          RDR07800
      IF (TAUOK .GT. 0.0) GO TO 6                                       RDR07900
C     DEFAULT TAUOK                                                     RDR08000
      TAUOK = 600.0                                                     RDR08100
```

```
    6 CONTINUE                                                          RDR08200
    8 DO 16 I=1,NNZ                                                     RDR08300
C       DEFAULT SIGYO                                                   RDR08400
      IF (SIGYO(I) .GT. 0.0) GO TO 9                                    RDR08500
      SIGYO(I) = SIGXO(I)                                               RDR08600
    9 CONTINUE                                                          RDR08700
      IF (ALPHA(I) .GT. 0.0) GO TO 10                                   RDR08800
C       DEFAULT ALPHA                                                   RDR08900
      ALPHA(I) = 1.0                                                    RDR09000
   10 IF (BETA(I) .GT. 0.0) GO TO 12                                    RDR09100
C       DEFAULT BETA                                                    RDR09200
      BETA(I) = 1.0                                                     RDR09300
   12 CONTINUE                                                          RDR09400
      IF (IZMOD(I) .GT. 0) GO TO 16                                     RDR09500
C       DEFAULT IZMOD                                                   RDR09600
      IZMOD(I) = 1                                                      RDR09700
   16 CONTINUE                                                          RDR09800
      IF (XRY .GT. 0.0) GO TO 18                                        RDR09900
C       DEFAULT XRY                                                     RDR10000
      XRY = 100.0                                                       RDR10100
   18 IF (XRZ .GT. 0.0) GO TO 20                                        RDR10200
C       DEFAULT XRZ                                                     RDR10300
      XRZ = 100.0                                                       RDR10400
   20 IF (TIMAV .GT. 0.0) GO TO 24                                      RDR10500
C       DEFAULT TIMAV                                                   RDR10600
      TIMAV = 600.0                                                     RDR10700
      IF (ISKIP(5) .EQ. 2) TIMAV = 360.0                                RDR10800
   24 IF (ZRK .GT. 0.0) GO TO 26                                        RDR10900
C       DEFAULT ZRK                                                     RDR11000
      ZRK = 2.0                                                         RDR11100
   26 CONTINUE                                                          RDR11200
      IF (ISKIP(6) .EQ. 0) ISKIP(6) = 2                                 RDR11300
C       CHECK IZMOD                                                     RDR11400
      KSW(2) = 0                                                        RDR11500
      NBK = 0                                                           RDR11600
      KSW(1) = 0                                                        RDR11700
      DO 34 I=1,NNZ                                                     RDR11800
      I1 = IZMOD(I)/100                                                 RDR11900
      I2 = (IZMOD(I)-I1*100)/10                                         RDR12000
      I3 = IZMOD(I)-I1*100-I2*10                                        RDR12100
      IF (I .GT. 1) GO TO 27                                            RDR12200
```

```
      IF (I1 .NE. 6.AND.I2 .NE. 6.AND.I3 .NE. 6) GO TO 27               RDR12300
      KSW(2) = 1                                                        RDR12400
      IF (ISKIP(1) .EQ. 0) ISKIP(1) = 1                                 RDR12500
      GO TO 72                                                          RDR12600
   27 IF (I1.NE.5.AND.I2.NE.5.AND.I3.NE.5) GO TO 28                     RDR12700
      ZLIM = Z(I+1)                                                     RDR12800
      KSW(1) = 1                                                        RDR12900
      II = I                                                            RDR13000
   28 IF (I2.EQ.9.OR.I1.EQ.9.OR.I3 .EQ. 9) GO TO 29                     RDR13100
      IF (I2.NE.4.AND.I1.NE.4.AND.I3 .NE. 4) GO TO 31                   RDR13200
      IF (NBK .GT. 0) GO TO 30                                          RDR13300
   29 NBK = NBK+1                                                       RDR13400
      JBOT(NBK) = I                                                     RDR13500
   30 JTOP(NBK) = I                                                     RDR13600
   31 NTAL = 0                                                          RDR13700
      MODLS(I) = 1                                                      RDR13800
   32 NTAL = NTAL+1                                                     RDR13900
      IF (NTAL .GT. 3) GO TO 33                                         RDR14000
      IF (I1 .EQ. NTAL.OR.I2 .EQ. NTAL.OR.I3 .EQ. NTAL) GO TO 33        RDR14100
      GO TO 32                                                          RDR14200
   33 IF (NTAL .LT. 4) MODLS(I) = NTAL                                  RDR14300
   34 CONTINUE                                                          RDR14400
      IF (KSW(1) .NE. 1) GO TO 72                                       RDR14500
      IF (ISKIP(1) .EQ. 0) ISKIP(1) = 1                                 RDR14600
      ISKIP(2) = 0                                                      RDR14700
      ISKIP(3) = 0                                                      RDR14800
      NPTS = II                                                         RDR14900
      DO 70 I=1,II                                                      RDR15000
   70 ZZL(I) = Z(I)                                                     RDR15100
      GO TO 73                                                          RDR15200
   72 IF (NPTS .GT. 0) GO TO 73                                         RDR15300
      NPTS = 1                                                          RDR15400
      ZZL(1) = 0.0                                                      RDR15500
   73 CONTINUE                                                          RDR15600
      IF (LAMBDA .LE. 0.0) GO TO 74                                     RDR15700
      IF (ZLIM .LE. 0.0) ZLIM = Z(NZS)                                  RDR15800
   74 CONTINUE                                                          RDR15900
      DO 36 I=1,NZS                                                     RDR16000
C     CHECK MINIMUM LIMITS                                              RDR16100
      IF (SIGAK(I) .LT. .5) SIGAK(I) = .5                               RDR16200
      IF (SIGEK(I) .LT. .1) SIGEK(I) = .1                               RDR16300
```

```
      IF (UBARK(I) .LT. .1) UBARK(I) = .1                                RDR16400
   36 CONTINUE                                                           RDR16500
      IF (NBK .EQ. 0) GO TO 57                                           RDR16600
      IF (ISKIP(7) .GT. 0) GO TO 40                                      RDR16700
C     DETERMINE LAYER CHANGE PARAMETERS                                  RDR16800
      ZRL = ZRK                                                          RDR16900
      II = -1                                                            RDR17000
      DO 38 I=1,NBK                                                      RDR17100
      II = II+2                                                          RDR17200
      NTAL = JBOT(I)                                                     RDR17300
      NTAK = JTOP(I)                                                     RDR17400
      UBARL(II) = UBARK(NTAL)                                            RDR17500
      UBARL(II+1) = UBARK(NTAK+1)                                        RDR17600
      SIGAL(II) = SIGAK(NTAL)                                            RDR17700
      SIGAL(II+1) = SIGAK(NTAK+1)                                        RDR17800
      SIGEL(II) = SIGEK(NTAL)                                            RDR17900
      SIGEL(II+1) = SIGEK(NTAK+1)                                        RDR18000
      THETAL(II) = THETAK(NTAL)                                          RDR18100
      THETAL(II+1) = THETAK(NTAK+1)                                      RDR18200
      ALPHL(I) = ALPHA(NTAL)                                             RDR18300
      BETL(I) = BETA(NTAL)                                               RDR18400
      TEMPL(II) = TEMPK(NTAL)                                            RDR18500
      TEMPL(II+1) = TEMPK(NTAK+1)                                        RDR18600
   38 CONTINUE                                                           RDR18700
      TAUOL = TAUOK                                                      RDR18800
      TAUL = TAUK                                                        RDR18900
      GO TO 52                                                           RDR19000
   40 CONTINUE                                                           RDR19100
      IF (TAUOL .GT. 0.0) GO TO 42                                       RDR19200
C     DEFAULT TAUOL                                                      RDR19300
      TAUOL = 600.0                                                      RDR19400
   42 IF (ZRL .GT. 0.0) GO TO 44                                         RDR19500
C     DEFAULT ZRL                                                        RDR19600
      ZRL = ZRK                                                          RDR19700
   44 DO 48 I=1,NBK                                                      RDR19800
      IF (ALPHL(I) .GT. 0.0) GO TO 46                                    RDR19900
C     DEFAULT ALPHL                                                      RDR20000
      ALPHL(I) = 1.0                                                     RDR20100
   46 IF (BETL(I) .GT. 0.0) GO TO 48                                     RDR20200
C     DEFAULT BETL                                                       RDR20300
      BETL(I) = 1.0                                                      RDR20400
```

```
   48 CONTINUE
      NTAL = 2*NBK
      DO 50 I=1,NTAL
C     CHECK MINIMUM VALUES
      IF (SIGAL(I) .LT. .5) SIGAL(I) = .5
      IF (UBARL(I) .LT. .1) UBARL(I) = .1
      IF (SIGEL(I) .LT. .1) SIGEL(I) = .1
   50 CONTINUE
   52 NTAK = NNZ+1
      NTAL = NNZ+NBK
C     COMBINE ALPHA AND BETA WITH ALPHL AND BETL
      DO 54 I=NTAK,NTAL
      INNZ = I-NNZ
      ALPHA(I) = ALPHL(INNZ)
      BETA(I) = BETL(INNZ)
   54 CONTINUE
   57 CONTINUE
   58 CONTINUE
      STO1 = (TAUK/TAUOK)**(0.2)*RAD
      S = (Z(2)/ZRK)
      S1 = 1.0/ALOG(S)
      P = RB8(UBARK(2),UBARK(1),S1)
C     CALCULATE UBAR FOR LAYER 1
      UBAR(1) = RB11(UBARK(1),P,Z(2),ZRK)
       PPWR = P
      IF (NNZ .LT. 2) GO TO 152
      DO 150 I=2,NNZ
C     CALCULATE UBAR FOR LAYERS 2 TO NNZ
  150 UBAR(I) = 0.5*(UBARK(I+1)+UBARK(I))
  152 P = RB8(SIGAK(2),SIGAK(1),S1)
C     CALCULATE SIGAP FOR LAYER 1
      SIGAP(1) = STO1*RB11(SIGAK(1),P,Z(2),ZRK)
      MPWR = P
      IF (NNZ .LT. 2) GO TO 162
      DO 160 I=2,NNZ
C     CALCULATE SIGAP FOR LAYERS 2 TO NNZ
  160 SIGAP(I) = 0.5*STO1*(SIGAK(I+1)+SIGAK(I))
  162 P = RB8(SIGEK(2),SIGEK(1),S1)
C     CALCULATE SIGEP FOR LAYER 1
      SIGEP(1) = RB11(SIGEK(1),P,Z(2),ZRK)*RAD
      IF (NNZ .LT. 2) GO TO 172
```

```
      QPWR = P                                                          RDR24600
      DO 170 I=2,NNZ                                                    RDR24700
C     CALCULATE SIGEP FOR LAYERS 2 TO NNZ                               RDR24800
  170 SIGEP(I) = ((SIGEK(I+1)+SIGEK(I))*RAD)*0.5                        RDR24900
  172 DO 180 I=1,NNZ                                                    RDR25000
      J = I                                                             RDR25100
C     CALCULATE THETA FOR ALL LAYERS                                    RDR25200
      THETA(I) = 0.5*(THETAK(J+1)+THETAK(J))                            RDR25300
      IF (ABS(THETAK(J+1)-THETAK(J)) .GT. 180.0) THETA(I) = THETA(I)-   RDR25400
     1180.0                                                             RDR25500
C     CALCULATE DELTHP FOR ALL LAYERS                                   RDR25600
      DELTHP(I) = THETAK(J+1)-THETAK(J)                                 RDR25700
      IF (DELTHP(I) .GT. 180.0) DELTHP(I) = 360.0-DELTHP(I)             RDR25800
      IF (DELTHP(I) .LT. -180.0) DELTHP(I) = 360.0+DELTHP(I)            RDR25900
  180 CONTINUE                                                          RDR26000
      DO 185 I=1,NNZ                                                    RDR26100
C     CALCULATE DELU FOR ALL LAYERS                                     RDR26200
      DELU(I) = UBARK(I+1)-UBARK(I)                                     RDR26300
      IF (DELU(I) .GE. 0.0) GO TO 185                                   RDR26400
      IF (TEMPK(I+1)-TEMPK(I) .GE. 0.0) GO TO 185                       RDR26500
      DELU(I) = ABS(DELU(I))                                            RDR26600
  185 CONTINUE                                                          RDR26700
      IF (KSW(2) .GT. 0) GO TO 250                                      RDR26800
      IF (NBK .EQ. 0) GO TO 250                                         RDR26900
      STO1 = (TAUL/TAUOL)**(0.2)*RAD                                    RDR27000
      M = JTOP(1)                                                       RDR27100
      IF (JBOT(1) .GT. 1) GO TO 186                                     RDR27200
      S = (Z(M+1)/ZRL)                                                  RDR27300
      S1 = 1.0/ALOG(S)                                                  RDR27400
  186 IF (ISKIP(7) .GT. 0) GO TO 192                                    RDR27500
      DO 188 I=1,NBK                                                    RDR27600
      NNZI = NNZ+I                                                      RDR27700
      M1 = JBOT(I)                                                      RDR27800
      M2 = JTOP(I)                                                      RDR27900
      S = 0.0                                                           RDR28000
      DO 187 J=M1,M2                                                    RDR28100
  187 S = S+0.5*(UBARK(J)+UBARK(J+1))*(Z(J+1)-Z(J))                     RDR28200
      UBAR(NNZI) = S/(Z(M2+1)-Z(M1))                                    RDR28300
  188 CONTINUE                                                          RDR28400
      GO TO 202                                                         RDR28500
  192 CONTINUE                                                          RDR28600
```

```
      IF (JBOT(1) .GT. 1) GO TO 193                                     RDR28700
      P = RB8(UBARL(2),UBARL(1),S1)                                     RDR28800
C     CALCULATE UBAR FOR NEW LAYER 1 (IF CONTAINS SURFACE)              RDR28900
      UBAR(NNZ+1) = RB11(UBARL(1),P,Z(M+1),ZRL)                         RDR29000
      QPWR = P                                                          RDR29100
      GO TO 197                                                         RDR29200
C     CALCULATE UBAR FOR NEW LAYER 1 (IF DOES'NT CONTAIN SURFACE)       RDR29300
  193 UBAR(NNZ+1) = (UBARL(1)+UBARL(2))*0.5                             RDR29400
  197 IF (NBK .LT. 2) GO TO 202                                         RDR29500
      DO 200 I=2,NBK                                                    RDR29600
      J = I*2-1                                                         RDR29700
C     CALCULATE UBAR FOR NEW LAYERS 2 TO NBK                            RDR29800
      NNZI = NNZ+I                                                      RDR29900
  200 UBAR(NNZI) = (UBARL(J+1)+UBARL(J))*0.5                            RDR30000
  202 IF (JBOT(1) .GT. 1) GO TO 210                                     RDR30100
      P = RB8(SIGEL(2),SIGEL(1),S1)                                     RDR30200
C     CALCULATE SIGEP FOR NEW LAYER 1 (IF CONTAINS SURFACE)             RDR30300
      SIGEP(NNZ+1) = RB11(SIGEL(1),P,Z(M+1),ZRL)*RAD                    RDR30400
      GO TO 215                                                         RDR30500
C     CALCULATE SIGEP FOR NEW LAYER 1 (IF DOES'NT CONTAIN SURFACE)      RDR30600
  210 SIGEP(NNZ+1) = ((SIGEL(2)+SIGEL(1))*RAD)*0.5                      RDR30700
  215 IF (NBK .LT. 2) GO TO 222                                         RDR30800
      DO 220 I=2,NBK                                                    RDR30900
      J = I*2-1                                                         RDR31000
C     CALCULATE SIGEP FOR NEW LAYERS 2 TO NBK                           RDR31100
      NNZI = NNZ+I                                                      RDR31200
  220 SIGEP(NNZI) = ((SIGEL(J+1)+SIGEL(J))*RAD)*0.5                     RDR31300
  222 IF (ISKIP(7) .GT. 0) GO TO 226                                    RDR31400
      DO 225 I=1,NBK                                                    RDR31500
C     CALCULATE THETA FOR NEW LAYERS 1 TO NBK                           RDR31600
      NNZI = NNZ+I                                                      RDR31700
      M1 = JBOT(I)                                                      RDR31800
      M2 = JTOP(I)                                                      RDR31900
      S = 0.0                                                           RDR32000
      DO 224 J=M1,M2                                                    RDR32100
      P = 0.5*(THETAK(J+1)+THETAK(J))                                   RDR32200
      IF (ABS(THETAK(J+1)-THETAK(J)) .LT. 180.0) GO TO 226              RDR32300
      P = P+180.0                                                       RDR32400
  224 S = S+P*(Z(J+1)-Z(J))                                             RDR32500
      THETA(NNZI) = S/(Z(M2+1)-Z(M1))                                   RDR32600
      DELTHP(NNZI) = THETAK(M2+1)-THETAK(M1)                            RDR32700
```

```
      IF (DELTHP(NNZI) .GT. 180.0) DELTHP(NNZI) = 360.0-DELTHP(NNZI)     RDR32800
      IF (DELTHP(NNZI) .LT. -180.0) DELTHP(NNZI) = 360.0+DELTHP(NNZI)    RDR32900
  225 CONTINUE                                                            RDR33000
      GO TO 230                                                           RDR33100
  226 DO 227 I=1,NBK                                                      RDR33200
      J = 2*I-1                                                           RDR33300
      NNZI = NNZ+I                                                        RDR33400
      THETA(NNZI) = 0.5*(THETAL(J+1)+THETAL(J))                           RDR33500
      IF(ABS(THETAL(J+1)-THETAL(J)).GT.180.)THETA(NNZI)=THETA(NNZI)+180.RDR33600
      DELTHP(NNZI) = THETAL(J+1)-THETAL(J)                                RDR33700
      IF (DELTHP(NNZI) .GT. 180.0) DELTHP(NNZI) = 360.0-DELTHP(NNZI)     RDR33800
      IF (DELTHP(NNZI) .LT. -180.0) DELTHP(NNZI) = 360.0+DELTHP(NNZI)    RDR33900
  227 CONTINUE                                                            RDR34000
  230 CONTINUE                                                            RDR34100
      DO 235 I=1,NBK                                                      RDR34200
      J = I*2-1                                                           RDR34300
C     CALCULATE DELTHP FOR ALL NEW LAYERS                                 RDR34400
      NNZI = NNZ+I                                                        RDR34500
      DELU(NNZI) = UBARL(J+1)-UBARL(J)                                    RDR34600
      IF (DELU(NNZI) .GE. 0.0) GO TO 235                                  RDR34700
      IF (TEMPL(J+1)-TEMPL(J) .GT. 0.0) GO TO 235                         RDR34800
      DELU(NNZI) = ABS(DELU(NNZI))                                        RDR34900
  235 CONTINUE                                                            RDR35000
  237 IF (JBOT(1) .GT. 1) GO TO 242                                       RDR35100
      P = RB8(SIGAL(2),SIGAL(1),S1)                                       RDR35200
C     CALCULATE SIGAP FOR NEW LAYER 1 (IF CONTAINS SURFACE)               RDR35300
      SIGAP(NNZ+1) = STO1*RB11(SIGAL(1),P,Z(M+1),ZRL)                     RDR35400
      GO TO 243                                                           RDR35500
C     CALCULATE SIGAP FOR NEW LAYER 1 (IF DOES'NT CONTAIN SURFACE)        RDR35600
  242 SIGAP(NNZ+1) = 0.5*STO1*(SIGAL(1)+SIGAL(2))                         RDR35700
  243 CONTINUE                                                            RDR35800
      IF (NBK .LT. 2) GO TO 250                                           RDR35900
      DO 245 I=2,NBK                                                      RDR36000
      J = I*2-1                                                           RDR36100
C     CALCULATE SIGAP FOR NEW LAYERS 2 TO NBK                             RDR36200
      NNZI = NNZ+I                                                        RDR36300
  245 SIGAP(NNZI) = 0.5*STO1*(SIGAL(J+1)+SIGAL(J))                        RDR36400
  250 CONTINUE                                                            RDR36500
      DO 395 I=2,NZS                                                      RDR36600
      IF (H .LE. Z(I)) GO TO 396                                          RDR36700
  395 CONTINUE                                                            RDR36800
```

```
  396 RNG = DELX(I-1)                                                    RDR36900
      AZM = DELY(I-1)                                                    RDR37000
CCC   GET DATE AND TIME (UNIVAC 1108 ONLY)                               RDR37100
      CALL ERTRAN(9,NTFB(1),NTFB(2))                                     RDR37200
C     LOAD MM/DD/YY INTO IDATE(1) AND (2) ON FIRST LOOP                  RDR37300
C     LOAD HR:MN:SC INTO ITIME(1) AND (2) ON SECOND LOOP                 RDR37400
      N = '/'                                                            RDR37500
      DO 400 I=1,2                                                       RDR37600
      J = 2*I-1                                                          RDR37700
      CALL MSFLD(0,12,NTFB(I),0,IDATE(J))                                RDR37800
      CALL MSFLD(0,6,N,12,IDATE(J))                                      RDR37900
      CALL MSFLD(12,12,NTFB(I),18,IDATE(J))                              RDR38000
      CALL MSFLD(0,6,N,30,IDATE(J))                                      RDR38100
      CALL MSFLD(24,12,NTFB(I),0,IDATE(J+1))                             RDR38200
  400 CONTINUE                                                           RDR38300
CCC   END DATE AND TIME - PRINT WITH A6,A2 FORMAT                        RDR38400
      WRITE (6,1000) NAMCAS,(TESTNO(I),I=1,6),IDATE,ITIME,H,RNG,AZM      RDR38500
      IF (ISKIP(8) .EQ. 1) WRITE (6,NAM2)                                RDR38600
      RETURN                                                             RDR38700
 1000 FORMAT (1H1,11(/),24X,21(4H****)/24X,1H*,82X,1H*/24X,1H*,5X,12A6,5RDR38800
     1X,1H*/24X,1H*,82X,1H*/24X,1H*,23X,6A6,23X,1H*/24X,1H*,82X,1H*/     RDR38900
     1                            24X,1H*,25X,7HDATE = ,A6,A2,9H, TIME = ,RDR39000
     2A6,A2,25X,1H*,3(/24X,1H*,82X,1H*)/24X,30H* ADJUSTED CLOUD RISE HEIRDR39100
     36HT =,F8.2,9H, RANGE =,F9.2,19H, AZIMUTH BEARING =,F7.2,2H */24X,1RDR39200
     4H*,82X,1H*/24X,21(4H****)/1H1)                                     RDR39300
      END                                                                RDR39400
```

```
      SUBROUTINE SGP(ZH,N,SIG,IN,IZ,UBHK,X,IBB)                         SGP00100
      COMMON /PARAMT/ TESTNO(12),         ISKIP(15),NXS,NYS,NZS,NDI,NCI, SGP00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),      SGP00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),        SGP00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,        SGP00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10),   SGP00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),SGP00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)SGP00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                    SGP00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),SGP01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,      SGP01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR,   SGP01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,       SGP01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,             SGP01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),          SGP01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,      SGP01600
     7NYSS,CDAMX(3)                                                        SGP01700
      DIMENSION DTHK(21)                                                   SGP01800
      EQUIVALENCE (DTHK,XAST)                                              SGP01900
      INTEGER TESTNO                                                       SGP02000
      REAL MPWR,L,LAMBDA                                                   SGP02100
C     SUBROUTINE SGP CALCULATES SIGENK AND SIGANK WITH OR WITHOUT          SGP02200
C     DESTRUCT IN THE LAYER.                                               SGP02300
      GO TO (4,44,64,68),IBB                                               SGP02400
    4 S = 0.0                                                              SGP02500
      MN = N-1                                                             SGP02600
      HHNK = ZH                                                            SGP02700
      HHRK = 1.0                                                           SGP02800
      IF (N .EQ. 1) GO TO 5                                                SGP02900
      HHRK = ZH                                                            SGP03000
      HHNK = Z(N+1)                                                        SGP03100
    5 SG3 = SIGEK(1)                                                       SGP03200
      IF (IN .EQ. 2) SG3 = SIGAP(1)                                        SGP03300
      IF (N .LE. 2) GO TO 30                                               SGP03400
      DO 25 M=2,MN                                                         SGP03500
      IF (IN .EQ. 2) GO TO 10                                              SGP03600
      SG1 = SIGEK(M+1)                                                     SGP03700
      SG2 = SIGEK(M)                                                       SGP03800
      GO TO 20                                                             SGP03900
   10 SG1 = SIGAP(M+1)                                                     SGP04000
```

```
      SG2 = SIGAP(M)                                                    SGP04100
   20 S = S+(SG1+SG2)*(Z(M+1)-Z(M))*0.5                                 SGP04200
   25 CONTINUE                                                          SGP04300
   30 IF (IN .EQ. 2) GO TO 35                                           SGP04400
      SG1 = SIGEK(N+1)                                                  SGP04500
      SG2 = SIGEK(N)                                                    SGP04600
      PWR = QPWR                                                        SGP04700
      GO TO 40                                                          SGP04800
   35 SG1 = SIGAP(N+1)                                                  SGP04900
      SG2 = SIGAP(N)                                                    SGP05000
      PWR = MPWR                                                        SGP05100
   40 IF (N .EQ. 1) GO TO 42                                            SGP05200
      S = S+(ZH-Z(N))*(((SG1-SG2)/(Z(N+1)-Z(N)))*(ZH-Z(N))+SG2)*0.5     SGP05300
   42 SIG = (S+RB11(SG3,PWR,HHNK,ZRK))*RAD/HHRK                         SGP05400
      RETURN                                                            SGP05500
C     ENTRY UBARS(ZH,N,IZ,UBHK)                                         SGP05600
C     SUBROUTINE UBARS CALCULATES UBARNK, X NK, Y NK, CAP THETA (ANG)   SGP05700
   44 XBARX = 0.0                                                       SGP05800
      YBARY(IZ) = 0.0                                                   SGP05900
      VV = VS(II)                                                       SGP06000
      PWR = PPWR                                                        SGP06100
      IF (JF .EQ. 2) VV = VB(II)                                        SGP06200
      IF (N .EQ. 1) GO TO 50                                            SGP06300
      MN = N-1                                                          SGP06400
      DO 45 M=1,MN                                                      SGP06500
      S = DTHK(M+1)-DTHK(M)                                             SGP06600
      IF (S) 46,45,46                                                   SGP06700
   46 CONTINUE                                                          SGP06800
      S1 = SIN(DTHK(M+1))-SIN(DTHK(M))                                  SGP06900
      S2 = COS(DTHK(M+1))-COS(DTHK(M))                                  SGP07000
      S = UBAR(M)*(Z(M+1)-Z(M))/(VV*S)                                  SGP07100
      XBARX = XBARX+(S1*S)                                              SGP07200
      YBARY(IZ) = YBARY(IZ)+(S2*(-S))                                   SGP07300
   45 CONTINUE                                                          SGP07400
   50 TMPQ1 = 1.0/(Z(N+1)-Z(N))                                         SGP07500
      S = ((DTHK(N+1)-DTHK(N))*TMPQ1)*(ZH-Z(N))+DTHK(N)                 SGP07600
      S1 = SIN(S)-SIN(DTHK(N))                                          SGP07700
      S2 = COS(S)-COS(DTHK(N))                                          SGP07800
      IF (N .EQ. 1) GO TO 52                                            SGP07900
      UBHK = ((UBARK(N+1)-UBARK(N))*TMPQ1)*0.5*(ZH-Z(N))+(0.5*UBARK(N)) SGP08000
      GO TO 54                                                          SGP08100
```

```
   52 UBHK = RB11(UBARK(1),PWR,ZH,ZRK)                                    SGP08200
   54 CONTINUE                                                            SGP08300
      S = DTHK(N+1)-DTHK(N)                                               SGP08400
      IF (S) 53,55,53                                                     SGP08500
   53 S = UBHK/(VV*S*TMPQ1)                                               SGP08600
      XBARX = XBARX+(S1*S)                                                SGP08700
      YBARY(IZ) = YBARY(IZ)+(S2*(-S))                                     SGP08800
   55 CONTINUE                                                            SGP08900
      IF (XBARX) 57,56,57                                                 SGP09000
   56 IF (YBARY(IZ)) 57,58,57                                             SGP09100
   57 ANG(IZ) = ATAN2(YBARY(IZ),XBARX)                                    SGP09200
      GO TO 60                                                            SGP09300
   58 ANG(IZ) = 0.0                                                       SGP09400
   59 UBARNK(IZ) = UBHK                                                   SGP09500
      SQBAR = 0.0                                                         SGP09600
      GO TO 62                                                            SGP09700
   60 IF (XBARX) 61,59,61                                                 SGP09800
   61 SQBAR = SQRT(XBARX*XBARX+YBARY(IZ)*YBARY(IZ))                       SGP09900
      UBARNK(IZ) = SQBAR*VV/ZH                                            SGP10000
   62 CONTINUE                                                            SGP10100
      RETURN                                                              SGP10200
C     ENTRY DEPSO(X,N,IZ)                                                 SGP10300
C     SUBROUTINE DEPSO CALCULATES ALL OF THE DEPOSITION EQUATION EXCEPT   SGP10400
C     THE LATERAL TERM                                                    SGP10500
   64 ZF = ZZL(IZ)                                                        SGP10600
      VV = VS(II)                                                         SGP10700
      GAMMB = GAMMA(II)                                                   SGP10800
      XXX = X                                                             SGP10900
      PERK = PERC(II)                                                     SGP11000
      IF (JF .EQ. 1) GO TO 165                                            SGP11100
      ZF = HB                                                             SGP11200
      VV = VB(II)                                                         SGP11300
      XXX = X+(SIGZO(N)/SIGENK(IZ))**(1.0/BETANK(IZ))                     SGP11400
      PERK = PERCB(II)                                                    SGP11500
  165 XKNK = 0.0                                                          SGP11600
      IF (GAMMB .GE. 1.0) GO TO 69                                        SGP11700
      S1 = 1.0/(SIGENK(IZ)*XXX**BETANK(IZ))                               SGP11800
      S2 = VV*XXX/UBARNK(IZ)                                              SGP11900
      S3 = -0.5*S1*S1                                                     SGP12000
      S4 = BETANK(IZ)*(S2-ZF)-S2                                          SGP12100
      S2 = S2-ZF                                                          SGP12200
```

```
      B = 1.0                                                           SGP12300
      XKNK = -S4*EXP(S2*S2*S3)                                          SGP12400
      A = 0.0                                                           SGP12500
   65 A = A+2.0                                                         SGP12600
      S5 = A*Z(N+1)                                                     SGP12700
      S6 = S5-S2                                                        SGP12800
      S7 = S5+S2                                                        SGP12900
      S7 = S7*S7*S3                                                     SGP13000
      S6 = S6*S6*S3                                                     SGP13100
      IF (A .LE. 2.0) GO TO 66                                          SGP13200
      IF (S6 .LT. -10.0.AND.S7 .LT. -10.0) GO TO 67                     SGP13300
   66 S5 = S5*BETANK(IZ)                                                SGP13400
      XKNK = XKNK+B*((S5+S4)*EXP(S7)+GAMMB*(S5-S4)*EXP(S6))             SGP13500
      IF (GAMMB .LE. 0.0) GO TO 67                                      SGP13600
      B = B*GAMMB                                                       SGP13700
      GO TO 65                                                          SGP13800
   67 CONTINUE                                                          SGP13900
      XY = (SIGYO(N)/SIGANK(IZ))**(1.0/ALPHNK(IZ))                      SGP14000
      SIGYNK = SQRT((SIGANK(IZ)*(X+XY)**ALPHNK(IZ))**2+(SIGENK(IZ)*XXX**SGP14100
     1BETANK(IZ)*YBARY(IZ)/ZF)**2)                                      SGP14200
      IF (SIGYNK .LE. 0.0) GO TO 69                                     SGP14300
      DEP = Q(N)*PERK*(1.0-GAMMB)*S1*XKNK/(6.2831853*SIGYNK*FLOAT(NXCI)*SGP14400
     1XXX)                                                              SGP14500
   69 CONTINUE                                                          SGP14600
      RETURN                                                            SGP14700
C     ENTRY BETAK(ZH,N,IZ)                                              SGP14800
C     SUBROUTINE BETAK CALCULATES  BETA NK AND ALPHA NK                 SGP14900
   68 S1 = 0.0                                                          SGP15000
      S2 = 0.0                                                          SGP15100
      IF (N .EQ. 1) GO TO 90                                            SGP15200
      MN = N-1                                                          SGP15300
      DO 70 M=1,MN                                                      SGP15400
      S1 = S1+BETA(M)*(Z(M+1)-Z(M))                                     SGP15500
      S2 = S2+ALPHA(M)*(Z(M+1)-Z(M))                                    SGP15600
   70 CONTINUE                                                          SGP15700
      TMPQ1 = 1.0/ZH                                                    SGP15800
      TMPQ2 = ZH-Z(N)                                                   SGP15900
      BETANK(IZ) = (S1+BETA(N)*TMPQ2)*TMPQ1                             SGP16000
      ALPHNK(IZ) = (S2+ALPHA(N)*TMPQ2)*TMPQ1                            SGP16100
      GO TO 95                                                          SGP16200
   90 BETANK(IZ) = BETA(N)                                              SGP16300
```

```
      ALPHNK(IZ) = ALPHA(N)                                              SGP16400
   95 CONTINUE                                                           SGP16500
      RETURN                                                             SGP16600
      END                                                                SGP16700
```

```
      SUBROUTINE SIGMA(XP,M,MM)                                         SGA00100
      COMMON /PARAMT/ TESTNO(12),       ISKIP(15),NXS,NYS,NZS,NDI,NCI,  SGA00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     SGA00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      SGA00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     SGA00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), SGA00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),SGA00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)SGA00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  SGA00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),SGA01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    SGA01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, SGA01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    SGA01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          SGA01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       SGA01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   SGA01600
     7NYSS,CDAMX(3)                                                     SGA01700
      INTEGER TESTNO                                                    SGA01800
      REAL MPWR,L,LAMBDA                                                SGA01900
C      ***** THIS SUBROUTINE CALCULATES THE STANDARD DEVIATIONS OF X,Y,ZSGA02000
       X = XP                                                           SGA02100
       IF (MM .EQ. 2) X = XAST(M)                                       SGA02200
       MMM = 1                                                          SGA02300
       SIGZ = 0.0                                                       SGA02400
       SIGY = 0.0                                                       SGA02500
       SIGX = 0.0                                                       SGA02600
       N = MODLS(M)                                                     SGA02700
       GO TO (40,20,30),N                                               SGA02800
    20 SIGY = SIGYO(M)                                                  SGA02900
       SIGX = SIGXO(M)                                                  SGA03000
       GO TO 220                                                        SGA03100
    30 B3 = SIGEP(M)                                                    SGA03200
       B4 = BETA(M)                                                     SGA03300
    40 A1 = 1.0                                                         SGA03400
       A2 = SIGYO(M)                                                    SGA03500
       A3 = SIGAP(M)                                                    SGA03600
       A4 = ALPHA(M)                                                    SGA03700
       A5 = DELTHP(M)                                                   SGA03800
       A6 = SIGXO(M)                                                    SGA03900
       L = 0.0                                                          SGA04000
```

```
      IF (DELU(M) .LE. 0.0) GO TO 45                                  SGA04100
      L = 0.28*X*DELU(M)/UBAR(M)                                      SGA04200
   45 IF (MM .EQ. 1) GO TO 60                                         SGA04300
      N = 1                                                           SGA04400
      GO TO 60                                                        SGA04500
   50 T1 = (THETA(M)-THETA(JF))*RAD                                   SGA04600
      A1 = 1.0                                                        SGA04700
      T2 = SIN(T1)                                                    SGA04800
      T1 = COS(T1)                                                    SGA04900
      A2 = SQRT((SIGX*T2)**2+(SIGY*T1)**2)                            SGA05000
      A3 = SIGAP(JF)                                                  SGA05100
      A4 = ALPHA(JF)                                                  SGA05200
      A5 = DELTHP(JF)                                                 SGA05300
      A6 = SQRT((SIGX*T1)**2+(SIGY*T2)**2)                            SGA05400
      B3 = SIGEP(JF)                                                  SGA05500
      B4 = BETA(JF)                                                   SGA05600
      L = 0.0                                                         SGA05700
      IF (DELU(JF) .LE. 0.0) GO TO 60                                 SGA05800
      L = 0.28*X*DELU(JF)/UBAR(JF)                                    SGA05900
   60 IF (A4-1.0) 70,80,70                                            SGA06000
   70 A1 = 1.0/A4                                                     SGA06100
      IF (MMM .EQ. 2) GO TO 90                                        SGA06200
      IF (A2-A3*XRY) 80,80,90                                         SGA06300
   80 XY = A2/A3                                                      SGA06400
      GO TO 91                                                        SGA06500
   90 XY = A4*XRY*(A2/(A3*XRY))**A1+XRY*(1.0-A4)                      SGA06600
   91 IF (MMM .EQ. 1) XY = XY-XLRY                                    SGA06700
      IF (XY .LT. 0.0) XY = 0.0                                       SGA06800
      IF (A4-1.0) 110,100,110                                         SGA06900
  100 T1 = A3*(X+XY)                                                  SGA07000
      GO TO 120                                                       SGA07100
  110 T1 = (X+XY-XRY*(1.0-A4))/(XRY*A4)                               SGA07200
      IF (T1 .LE. 0.0) GO TO 125                                      SGA07300
      T1 = A3*XRY*T1**A4                                              SGA07400
  120 T2 = ABS(A5)*X*4.0589052E-3                                     SGA07500
      SIGY = SQRT(T1*T1+T2*T2)                                        SGA07600
  125 SIGX = SQRT(L*L*.05408329+A6*A6)                                SGA07700
      IF (N .EQ. 1) GO TO 220                                         SGA07800
      GO TO (150,130),MMM                                             SGA07900
  130 IF (B4-1.0) 140,131,140                                         SGA08000
  131 XZ = X                                                          SGA08100
```

```
      GO TO 190
  140 T1 = X/XRZ
      GO TO 210
  150 IF (B4-1.0) 151,160,151
  151 B1 = 1.0/B4
      IF (SIGZO(M)-B3*XRZ) 160,160,170
  160 XZ = SIGZO(M)/B3-XLRZ
      GO TO 180
  170 XZ = B4*XRZ*(SIGZO(M)/(B3*XRZ))**B1-XLRZ+XRZ*(1.0-B4)
  180 IF (XZ .LT. 0.0) XZ = 0.0
      XZ = X+XZ
      IF (B4-1.0) 200,190,200
  190 SIGZ = B3*XZ
      GO TO 220
  200 T1 = (XZ-XRZ*(1.0-B4))/(B4*XRZ)
      IF (T1 .LE. 0.0) GO TO 220
  210 SIGZ = B3*XRZ*T1**B4
  220 CONTINUE
      IF (MM .NE. 2) GO TO 240
      IF (MMM .EQ. 2) GO TO 230
      N = 2
      X = XP
      MMM = 2
      GO TO 50
  230 SIGXNK = SIGX
      SIGYNK = SIGY
  240 RETURN
      END
```

```
      SUBROUTINE TESTR(KTK)                                             TST00100
      COMMON /PARAMT/ TESTNO(12),         ISKIP(15),NXS,NYS,NZS,NDI,NCI, TST00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),      TST00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),       TST00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,      TST00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), TST00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),TST00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)TST00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                   TST00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),TST01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,     TST01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, TST01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,     TST01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,           TST01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),        TST01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,    TST01600
     7NYSS,CDAMX(3)                                                      TST01700
      INTEGER TESTNO                                                    TST01800
      REAL MPWR,L,LAMBDA                                                TST01900
C     THIS SUBROUTINE DETERMINES THE STRUCTURAL CHANGE IN LAYERS FOR    TST02000
C     THE PULL TRANSITION MODEL                                         TST02100
      IF (NBK .EQ. 0) GO TO 100                                         TST02200
      IF (KTK .EQ. 0) GO TO 50                                          TST02300
      IF (KK .GE. JBOT(ILK)) GO TO 50                                   TST02400
      IBOT = JBOT(ILK)                                                  TST02500
      ITOP = JTOP(ILK)                                                  TST02600
      GO TO 61                                                          TST02700
   50 IF (KK .NE. JBOT(ILK)) GO TO 61                                   TST02800
      IBOT = KK                                                         TST02900
      ITOP = JTOP(ILK)                                                  TST03000
      DO 60 J=IBOT,ITOP                                                 TST03100
   60 XAST(J) = UBAR(J)*TAST(ILK)                                       TST03200
      ILK = ILK+1                                                       TST03300
   61 CONTINUE                                                          TST03400
      KTK = 0                                                           TST03500
  100 CONTINUE                                                          TST03600
      RETURN                                                            TST03700
      END                                                               TST03800
```

```
      SUBROUTINE ISO(MR,MT)                                             IS000100
      COMMON /PARAMT/ TESTNO(12),        ISKIP(15),NXS,NYS,NZS,NDI,NCI, IS000200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     IS000300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      IS000400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     IS000500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), IS000600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),IS000700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)IS000800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  IS000900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),IS001000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    IS001100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, IS001200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    IS001300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          IS001400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       IS001500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   IS001600
     7NYSS,CDAMX(3)                                                     IS001700
      DIMENSION ERFX(6)                                                 IS001800
      EQUIVALENCE (ANG(10),ERFX)                                        IS001900
      INTEGER TESTNO                                                    IS002000
      REAL MPWR,L,LAMBDA                                                IS002100
      DOUBLE PRECISION A6,A7,A8,A9,A10,A11,DTX                          IS002200
C     THIS SUBROUTINE EVALUATES ERF(X)                                  IS002300
      DATA A6,A7,A8,A9,A10,A11/.0705230784D0,.0422820123D0,.0092705272D0IS002400
     1,.0001520143D0,.0002765672D0,.0000430638D0/                       IS002500
      DO 10 M=MR,MT                                                     IS002600
      IN = 0                                                            IS002700
      IF (ERFX(M) .LT. 0.0) IN = 1                                      IS002800
      ERFX(M) = ABS(ERFX(M))                                            IS002900
      IF (ERFX(M) .LT. 1.0E-10) GO TO 5                                 IS003000
      IF (ERFX(M) .GT. 5.0) GO TO 6                                     IS003100
      DTX = 1.0D0+ERFX(M)*(A6+ERFX(M)*(A7+ERFX(M)*(A8+ERFX(M)*(A9+ERFX(MIS003200
     1)*(A10+ERFX(M)*A11)))))                                           IS003300
      ERFX(M) = 1.0D0-(1.0D0/DTX)**16                                   IS003400
      GO TO 7                                                           IS003500
    5 ERFX(M) = 0.0                                                     IS003600
      GO TO 10                                                          IS003700
    6 ERFX(M) = 1.0                                                     IS003800
    7 IF (IN .EQ. 1) ERFX(M) = -ERFX(M)                                 IS003900
   10 CONTINUE                                                          IS004000
```

```
41*        RETURN                                            IS004100
42*        END                                               IS004200
```

```
      SUBROUTINE COORD(N,M,X,Y,XO,YO,ASP,XS,ICK)                        CRD00100
      COMMON /PARAMT/ TESTNO(12),        ISKIP(15),NXS,NYS,NZS,NDI,NCI, CRD00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     CRD00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      CRD00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     CRD00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), CRD00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VR(20),PERCB(20),CRD00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)CRD00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  CRD00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),CRD01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    CRD01100
     2STO2,STO3,TRU,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, CRD01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    CRD01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          CRD01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       CRD01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   CRD01600
     7NYSS,CDAMX(3)                                                     CRD01700
      INTEGER TESTNO                                                    CRD01800
      REAL MPWR,L,LAMBDA                                                CRD01900
C      *****THIS SUBROUTINE TRANSLATES AND ROTATES THE FIXED INPUT *****CRD02000
C      ***** COORDINATES RELATIVE TO A SYSTEM WITH POSITIVE X AXIS  ****CRD02100
C      ***** ALONG THE WIND DIRECTION THETA.                            CRD02200
      N = 0                                                             CRD02300
      B = AMOD(YO,360.0)*RAD                                            CRD02400
      IF (ICK .EQ. 2) GO TO 10                                          CRD02500
      A = THETA(M)                                                      CRD02600
      GO TO 11                                                          CRD02700
   10 A = THETA(JF)                                                     CRD02800
   11 XP = XO*SIN(B)                                                    CRD02900
      YP = XO*COS(B)                                                    CRD03000
      A = A*RAD                                                         CRD03100
      B = COS(A)                                                        CRD03200
      A = SIN(A)                                                        CRD03300
      DX = 0.0                                                          CRD03400
      DY = 0.0                                                          CRD03500
      DY = DELY(M)*RAD                                                  CRD03600
      DX = DELX(M)*SIN(DY)                                              CRD03700
      DY = DELX(M)*COS(DY)                                              CRD03800
   20 IF (ICK .EQ. 2) GO TO 50                                          CRD03900
   21 X1 = XP-DX                                                        CRD04000
```

```
      Y1 = YP-DY                                                        CRD04100
      X = -X1*A-Y1*B                                                    CRD04200
      Y = X1*B-Y1*A                                                     CRD04300
      IF (X .LE. 0.0) GO TO 80                                          CRD04400
      IF (KSW(2) .EQ. 0) GO TO 40                                       CRD04500
      XS = SQRT(X1*X1+Y1*Y1)                                            CRD04600
      ASP = 0.0                                                         CRD04700
      IF (X1) 31,30,31                                                  CRD04800
   30 IF (Y1) 31,90,31                                                  CRD04900
   31 ASP = 1.5707963-ATAN2(Y1,X1)                                      CRD05000
      IF (ASP .LT. 0.0) ASP = ASP+6.2831853                             CRD05100
      GO TO 90                                                          CRD05200
   40 IF (NBK .EQ. 0) GO TO 90                                          CRD05300
      IF (ICK .EQ. 2) GO TO 90                                          CRD05400
      IF (KK .LT. IBOT.OR.KK .GT. ITOP) GO TO 90                        CRD05500
      IF (XAST(M) .LE. 0.0) GO TO 80                                    CRD05600
      ASP = THETA(JF)*RAD                                               CRD05700
   50 XS = (THETA(M)+180.0)*RAD                                         CRD05800
      DX = DX+XAST(M)*SIN(XS)                                           CRD05900
      DY = DY+XAST(M)*COS(XS)                                           CRD06000
      IF (ICK .EQ. 2) GO TO 21                                          CRD06100
      X1 = XP-DX                                                        CRD06200
      Y1 = YP-DY                                                        CRD06300
      XS = -X1*SIN(ASP)-Y1*COS(ASP)                                     CRD06400
      A = ABS(THETA(M)-THETA(JF))                                       CRD06500
      IF (A .GE. 180.0) A = 360.0-A                                     CRD06600
      IF (A .GT. 45.0) GO TO 60                                         CRD06700
      IF (XS .LE. 0.0) GO TO 90                                         CRD06800
      GO TO 80                                                          CRD06900
   60 CALL SIGMA(XAST(M),M,3)                                           CRD07000
      ASP = A*RAD                                                       CRD07100
      SIGY = 2.15*SQRT((SIGX*SIN(ASP))**2+(SIGY*COS(ASP))**2)           CRD07200
      IF (A .GT. 90.0) GO TO 70                                         CRD07300
      IF (X .GT. XAST(M)+SIGY) GO TO 80                                 CRD07400
      IF (XS .LE. 0.0) GO TO 90                                         CRD07500
      IF (X .LT. XAST(M)) GO TO 90                                      CRD07600
      GO TO 80                                                          CRD07700
   70 IF (X .LE. XAST(M)+SIGY) GO TO 90                                 CRD07800
   80 N = 9                                                             CRD07900
   90 RETURN                                                            CRD08000
      END                                                               CRD08100
```

```
      SUBROUTINE PRTTTL(NWD,LINES,LINE,A,B)                             PRT00100
C     THIS SUBROUTINE PRINTS THE PAGE HEADING                           PRT00200
      DIMENSION LINE(1)                                                 PRT00300
      DATA IB/1H1/,JB/1H /                                              PRT00400
      N = IB                                                            PRT00500
      LINES = 3                                                         PRT00600
      N1= 1                                                             PRT00700
      N2 = 15                                                           PRT00800
   10 IF (N2 .GT. NWD) N2 = NWD                                         PRT00900
      WRITE (6,50) N,(LINE(I),I=N1,N2)                                  PRT01000
      LINES = LINES+1                                                   PRT01100
      N = JB                                                            PRT01200
      IF (N2 .GE. NWD) GO TO 20                                         PRT01300
      N1 = N2+1                                                         PRT01400
      N2 = N2+15                                                        PRT01500
      GO TO 10                                                          PRT01600
   20 IF (A .GE. 0.0) WRITE (6,80)                                      PRT01700
      LINES = LINES+1                                                   PRT01800
      IF (A .LE. 0.0) GO TO 30                                          PRT01900
      WRITE (6,60)                                                      PRT02000
      LINES = LINES+1                                                   PRT02100
   30 IF (B .LE. 0.0) GO TO 40                                          PRT02200
      WRITE (6,70)                                                      PRT02300
      LINES = LINES+1                                                   PRT02400
   40 RETURN                                                            PRT02500
   50 FORMAT (A1,19X,15A6)                                              PRT02600
   60 FORMAT (42X,45H(DECAY HAS BEEN INCLUDED IN THE CALCULATIONS))     PRT02700
   70 FORMAT (33X,64H(PRECIPITATION SCAVENGING HAS BEEN INCLUDED IN THE PRT02800
     1CALCULATIONS))                                                    PRT02900
   80 FORMAT ( )                                                        PRT03000
      END                                                               PRT03100
```

```
      SUBROUTINE PACKS(LINE,NWD)                                         PCK00100
C     THIS SUBROUTINE REMOVES EXCESSIVE BLANKS FROM THE TITLE AND PACKS PCK00200
C     IT INTO SUCCESSIVE LINES OF 15 WORDS PER LINE                      PCK00300
      DIMENSION LINE(1)                                                  PCK00400
      DATA IBLK/000000000005/                                            PCK00500
CC    DATA IBLK/000000000000/   - IBM 7044 -                             PCK00600
      NFLG = 0                                                           PCK00700
      NR = 0                                                             PCK00800
      JB = 16                                                            PCK00900
C     JB IS 15+1 OR 15 WORDS PER LINE                                    PCK01000
      IB = 1                                                             PCK01100
      LST = IBLK                                                         PCK01200
      M = 1                                                              PCK01300
      N = 0                                                              PCK01400
      J = 0                                                              PCK01500
   10 J = J+1                                                            PCK01600
      IF (J .LE. NWD) GO TO 15                                           PCK01700
      NFLG = 1                                                           PCK01800
      L = IBLK                                                           PCK01900
      N1 = N+1                                                           PCK02000
      IF (N1 .GT. 6) GO TO 80                                            PCK02100
      M1 = M                                                             PCK02200
      GO TO 60                                                           PCK02300
   15 K = LINE(J)                                                        PCK02400
      I = 0                                                              PCK02500
   20 I = I+1                                                            PCK02600
      IF (I .GT. 6) GO TO 10                                             PCK02700
      II = IABS(6*(I-1))                                                 PCK02800
      CALL MSFLD(II,6,K,30,L)                                            PCK02900
      IF (L .NE. IBLK) GO TO 30                                          PCK03000
      IF (LST .EQ. IBLK) GO TO 20                                        PCK03100
   25 IIBLK = I                                                          PCK03200
      JJBLK = J                                                          PCK03300
   30 N = N+1                                                            PCK03400
      NR = NR+1                                                          PCK03500
      IF (N .LT. 7) GO TO 40                                             PCK03600
      N = 1                                                              PCK03700
      IB = IB+1                                                          PCK03800
      M = M+1                                                            PCK03900
   40 II = IABS(6*(N-1))                                                 PCK04000
```

```
      IF (L .NE. IBLK) GO TO 50                                        PCK04100
      NNBLK = N                                                        PCK04200
      MMBLK = M                                                        PCK04300
   50 IF (IB .LT. JB) GO TO 70                                         PCK04400
      IF (LST .EQ. IBLK.OR.L .EQ. IBLK) GO TO 69                       PCK04500
      L = IBLK                                                         PCK04600
      NR = NR-1                                                        PCK04700
      N1 = NNBLK+1                                                     PCK04800
      M1 = MMBLK                                                       PCK04900
      IF (N1 .LT. 7) GO TO 60                                          PCK05000
      M1 = M1+1                                                        PCK05100
      N1 = 1                                                           PCK05200
   60 II = IABS(6*(N1-1))                                              PCK05300
      CALL MSFLD(30,6,L,II,LINE(M1))                                   PCK05400
      IF (NFLG .EQ. 1) GO TO 63                                        PCK05500
      GO TO 64                                                         PCK05600
   63 NR = NR+1                                                        PCK05700
   64 CONTINUE                                                         PCK05800
      N1 = N1+1                                                        PCK05900
      IF (N1 .LT. 7) GO TO 60                                          PCK06000
      IF (NFLG .EQ. 1) GO TO 80                                        PCK06100
      N1 = 1                                                           PCK06200
      M1 = M1+1                                                        PCK06300
      IF (M1 .LT. M) GO TO 60                                          PCK06400
      J = JJBLK                                                        PCK06500
      I = IIBLK                                                        PCK06600
      N = 7                                                            PCK06700
      M = M-1                                                          PCK06800
      LST = IBLK                                                       PCK06900
      IB = 0                                                           PCK07000
      K = LINE(J)                                                      PCK07100
      GO TO 20                                                         PCK07200
   69 IB = 1                                                           PCK07300
      LST = IBLK                                                       PCK07400
      IF (L .NE. IBLK) GO TO 70                                        PCK07500
      NR = NR-1                                                        PCK07600
      IB = 0                                                           PCK07700
      N = 7                                                            PCK07800
      M = M-1                                                          PCK07900
      GO TO 20                                                         PCK08000
   70 LST = L                                                          PCK08100
```

```
   75 CALL MSFLD(30,6,L,II,LINE(M))                                     PCK08200
      GO TO 20                                                          PCK08300
   80 NWD = NR/6                                                        PCK08400
      IB = NWD*6                                                        PCK08500
      IF (IB .LT. NR) NWD = NWD+1                                       PCK08600
      RETURN                                                            PCK08700
      END                                                               PCK08800
```

```
      SUBROUTINE HEDING(KSW,KS,JSW,LSW)                                 HED00100
C     SET ALL PARAMETERS NEC TO BUILDING PAGE HEADING                   HED00200
      DIMENSION KSW(1)                                                  HED00300
      KSW(3) = 3*JSW-3                                                  HED00400
      GO TO (10,40,50,60,70),KS                                         HED00500
   10 IF (KSW(1) .EQ. 0) GO TO 20                                       HED00600
      KSW(4) = 18                                                       HED00700
      GO TO 80                                                          HED00800
   20 IF (KSW(2) .EQ. 0) GO TO 30                                       HED00900
      KSW(4) = 30                                                       HED01000
      GO TO 80                                                          HED01100
   30 KSW(4) = 0                                                        HED01200
      IF (LSW .EQ. 1) KSW(4) = 24                                       HED01300
      GO TO 80                                                          HED01400
   40 KSW(4) = 6                                                        HED01500
      GO TO 80                                                          HED01600
   50 KSW(4) = 12                                                       HED01700
      GO TO 80                                                          HED01800
   60 KSW(4) = 36                                                       HED01900
      GO TO 80                                                          HED02000
   70 KSW(4) = 42                                                       HED02100
   80 RETURN                                                            HED02200
      END                                                               HED02300
```

```
      SUBROUTINE BOUNDS(X1,Y1,XLST,YLST)                                BND00100
C     CONFINE PLOT POINTS INSIDE OF AXES                                BND00200
      COMMON /BNDS/ XRIT,XLFT,YBOT,YTOP,XPL,YPL                         BND00300
      A = (YLST-Y1)/(XLST-X1)                                           BND00400
      B = Y1-A*X1                                                       BND00500
      IF (X1 .GT. XRIT) GO TO 90                                        BND00600
      IF (X1 .LT. XLFT) GO TO 60                                        BND00700
      IF (Y1 .LT. YBOT) GO TO 20                                        BND00800
      YPL = YTOP                                                        BND00900
   10 XPL = (YPL-B)/A                                                   BND01000
      GO TO 110                                                         BND01100
   20 IF (X1 .GT. XRIT) GO TO 50                                        BND01200
      IF (X1 .LT. XLFT) GO TO 30                                        BND01300
      YPL = YBOT                                                        BND01400
      GO TO 10                                                          BND01500
C     LOWER LEFT HAND CORNER ASSUME CROSSES XLFT                        BND01600
   30 XPL = XLFT                                                        BND01700
   40 YPL = A*XPL+B                                                     BND01800
      IF (YPL .GE. YBOT) GO TO 110                                      BND01900
C     WRONG CROSSES YBOT                                                BND02000
      YPL = YBOT                                                        BND02100
      GO TO 10                                                          BND02200
C     LOWER RIGHT HAND CORNER ASSUME CROSSES XRIT                       BND02300
   50 XPL = XRIT                                                        BND02400
      GO TO 40                                                          BND02500
   60 IF (Y1 .GT. YTOP) GO TO 70                                        BND02600
      IF (Y1 .LT. YBOT) GO TO 30                                        BND02700
      XPL = XLFT                                                        BND02800
      GO TO 40                                                          BND02900
C     UPPER LEFT HAND CORNER ASSUME CROSSES XLFT                        BND03000
   70 XPL = XLFT                                                        BND03100
   80 YPL = A*XPL+B                                                     BND03200
      IF (YPL .LE. YTOP) GO TO 110                                      BND03300
C     WRONG CROSSES YTOP                                                BND03400
      YPL = YTOP                                                        BND03500
      GO TO 10                                                          BND03600
   90 IF (Y1 .LT. YBOT) GO TO 50                                        BND03700
      IF (Y1 .GT. YTOP) GO TO 100                                       BND03800
      XPL = XRIT                                                        BND03900
      GO TO 40                                                          BND04000
```

```
      C     UPPER RIGHT HAND CORNER ASSUME CROSSES XRIT                          BND04100
        100 XPL = XRIT                                                           BND04200
            GO TO 80                                                             BND04300
        110 RETURN                                                               BND04400
            END                                                                  BND04500
```

```
      SUBROUTINE VRTCLE(KS,JM,KSW,ISTOR,ISKIP5,NCV)                     VRT00100
      DIMENSION KSW(1),ISTOR(1)                                         VRT00200
      COMMON /LBLLBL/ J1(9),J2(4),J3(48),J5(6),J7(3),J8(16),J9(13),J10, VRT00300
     1J4(12),J11(2),UNIT(15)                                            VRT00400
      INTEGER UNIT                                                      VRT00500
      IF (JM .GT. 1) GO TO 20                                           VRT00600
      I1 = 31                                                           VRT00700
      IF (KSW(2) .GT. 0) GO TO 10                                       VRT00800
      I1 = 19                                                           VRT00900
   10 I2 = I1+4                                                         VRT01000
      I3 = 8                                                            VRT01100
      GO TO 82                                                          VRT01200
   20 GO TO (30,40,50,60,70),KS                                         VRT01300
   30 I1 = 1                                                            VRT01400
      I2 = 3                                                            VRT01500
      GO TO 80                                                          VRT01600
   40 I1 = 7                                                            VRT01700
      I2 = 8                                                            VRT01800
      I3 = 3                                                            VRT01900
      GO TO 81                                                          VRT02000
   50 I1 = 13                                                           VRT02100
      I2 = 17                                                           VRT02200
      GO TO 80                                                          VRT02300
   60 I1 = 37                                                           VRT02400
      I2 = 40                                                           VRT02500
      I3 = 6                                                            VRT02600
      GO TO 82                                                          VRT02700
   70 I1 = 43                                                           VRT02800
      I2 = 47                                                           VRT02900
   80 I3 = 0                                                            VRT03000
   81 IF (ISKP5 .EQ. 4) I3 = I3+9                                       VRT03100
   82 ISTOR(1) = J3(4)                                                  VRT03200
      DO 90 I=I1,I2                                                     VRT03300
   90 ISTOR(I-I1+2) = J3(I)                                             VRT03400
      NCV = I2-I1+2                                                     VRT03500
      IF (JM .GT. 1) GO TO 110                                          VRT03600
      DO 100 I=1,2                                                      VRT03700
  100 ISTOR(NCV+I) = J4(I3+I)                                           VRT03800
      NCV = NCV+2                                                       VRT03900
      GO TO 130                                                         VRT04000
```

```
  110 DO 120 I=1,3                                          VRT04100
  120 ISTOR(NCV+I) = UNIT(I3+I)                             VRT04200
      NCV = NCV+3                                           VRT04300
  130 CALL PACKS(ISTOR,NCV)                                 VRT04400
      RETURN                                                VRT04500
      END                                                   VRT04600
```

```
      SUBROUTINE LLPLOT(YAR,XAR,N,TITLE,CRIT,NCRIT,NWD,VERTCL,NCV)      LLP00100
      COMMON /XYXYPT/ YP(41),XP(41),A(41),B(41),C(41),D(41),XI(41),YI(41LLP00200
     1),NUM(3),NC                                                       LLP00300
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, LLP00400
     1XSIZE1,YSIZE1                                                     LLP00500
      DIMENSION XAR(1),YAR(1),LINE(120),TITLE(1),CRIT(1),XF(1),YF(1),   LLP00600
     1FLDX(5),LEXP(3),VERTCL(1)                                         LLP00700
      COMMON /ILALPH/ LCRIT(10),IBLANK,ISTAR,IP1,IP2,IP3,HLABEL(5),NCH  LLP00800
      COMMON /PLTLLO/ ISW,XMAXJN,YMAXJN,XCIZE,YCIZE                     LLP00900
      DATA HLABEL/27HALONGWIND DISTANCE (METERS)/,NCH/27/               LLP01000
      DATA IBLANK/1H /,LCRIT/2HA=,2HB=,2HC=,2HD=,2HE=,2HF=,2HG=,2HH=,2HILLP01100
     1=,2HJ=/,ISTAR/1H*/                                                LLP01200
      EQUIVALENCE (FLDX,HLABEL),(LINE,B),(XF,XI),(YF,YI)                LLP01300
      EQUIVALENCE (XLM1,NND),(YBM1,NST),(HT,IPWRX),(CHARF,IPWRY),(SCLX,FLLP01400
     1II),(SCLY,II),(XSIZE1,IEXP),(YSIZE1,PWRY)                         LLP01500
      DATA LEXP/2H5-,2H2-,2H  /                                         LLP01600
 2010 FORMAT (7X,1H(,20(6H------),1H))                                  LLP01700
 2020 FORMAT (4X,3H10-,1H(,3(11(1H-),1HI,15(1H-),1HI,11(1H-),1HI),1H)/8XLLP01800
     1,3(I1,39X),I1/6X,3(2H10,11X,1H2,15X,1H5,10X),2H10/54X,5A6)        LLP01900
 2030 FORMAT (1X,A1,3X,A2,1H(,120A1,1H))                                LLP02000
 2040 FORMAT (1X,A1,3X,I2,1H(,120A1,1H))                                LLP02100
 2050 FORMAT (1X,A1,2X,4H10-(,120A1,1H))                                LLP02200
 2080 FORMAT (2(5(10X,A1,1H=,E10.3,1H,)/))                              LLP02300
 2090 FORMAT (86H0 ** NO PLOTS THIS CASE -- DOSAGE OR CONCENTRATION VALULLP02400
     1ES ARE PROBABLY OUT OF RANGE **//)                                LLP02500
C     FIND XMIN, XMAX, YMIN, YMAX                                       LLP02600
      YMAX = -1.E20                                                     LLP02700
      NND = N+1                                                         LLP02800
      NST = 0                                                           LLP02900
    5 NST = NST+1                                                       LLP03000
      IF (NST .GE. N) GO TO 500                                         LLP03100
      IF (XAR(NST) .LE. 0.0 .OR. YAR(NST) .LE. 0.0) GO TO 5             LLP03200
   10 NND = NND-1                                                       LLP03300
      IF (NND .LE. 1) GO TO 500                                         LLP03400
      IF (XAR(NND) .LE. 0.0 .OR. YAR(NND) .LE. 0.0) GO TO 10            LLP03500
      IF (NND .LE. NST) GO TO 500                                       LLP03600
      DO 15 I=NST,NND                                                   LLP03700
      IF (XAR(I) .LE. 0.0) GO TO 500                                    LLP03800
      XF(I) = XAR(I)                                                    LLP03900
      IF (ISW .NE. 2) XF(I) = ALOG10(XAR(I))                            LLP04000
```

```
      IF (YAR(I) .LE. 0.0) GQ TO 14                                    LLP04100
      YF(I) = ALOG10(YAR(I))                                           LLP04200
      IF (YMAX .LT. YF(I)) YMAX = YF(I)                                LLP04300
      GO TO 15                                                         LLP04400
   14 YF(I) = -1.E20                                                   LLP04500
   15 CONTINUE                                                         LLP04600
      IPWRX = INT(XF(NST)+100.0)-100                                   LLP04700
      IPWRY = INT(YMAX+100.0)-102                                      LLP04800
      IF (IPWRX .LT. 2) IPWRX = 2                                      LLP04900
      PWRY = FLOAT(IPWRY)                                              LLP05000
C     PRINT TITLE INFORMATION                                          LLP05100
      CALL PRTTTL(NWD,LINES,TITLE,-1.0,0.0)                            LLP05200
      WRITE (6,2010)                                                   LLP05300
C     LOOP FOR 48 PRINTER LINES                                        LLP05400
      JK1 = 0                                                          LLP05500
      JK2 = 1                                                          LLP05600
      JT = 0                                                           LLP05700
      IST = (48-NCV)/2                                                 LLP05800
      DO 220 I=1,48                                                    LLP05900
      IAI = LEXP(3)                                                    LLP06000
      IF (I .LT. IST) GO TO 17                                         LLP06100
      JT = JT+1                                                        LLP06200
      IF (JT .GT. NCV) GO TO 17                                        LLP06300
      JK1 = JK1+1                                                      LLP06400
      IF (JK1 .LT. 7) GO TO 16                                         LLP06500
      JK1 = 1                                                          LLP06600
      JK2 = JK2+1                                                      LLP06700
   16 CALL MSFLD(IABS(6*(JK1-1)),6,VERTCL(JK2),0,IAI)                  LLP06800
   17 CONTINUE                                                         LLP06900
      II=48-I                                                          LLP07000
      FII = FLOAT(II)*0.0625                                           LLP07100
      IF (NCRIT) 60,60,20                                              LLP07200
   20 IF (NCRIT .EQ. 9) GO TO 60                                       LLP07300
      DO 40 KK=1,NCRIT                                                 LLP07400
      IF (CRIT(KK) .LE. 0.0) GO TO 40                                  LLP07500
      IF (ABS(ALOG10(CRIT(KK))-FII-PWRY) .GT. 0.031255) GO TO 40       LLP07600
      DO 30 LL=1,120                                                   LLP07700
   30 LINE(LL) = LCRIT(KK)                                             LLP07800
      GO TO 67                                                         LLP07900
   40 CONTINUE                                                         LLP08000
   60 DO 65 J=1,120                                                    LLP08100
```

```
   65 LINE(J) = IBLANK                                                LLP08200
   67 DO 70 J=NST,NND                                                 LLP08300
      IF (ABS(YF(J)-FI1-PWRY) .GT. 0.031255) GO TO 70                 LLP08400
      L = INT(XF(J)*40.0+0.5)-40*IPWRX                                LLP08500
      IF (L .LT. 1 .OR. L .GT. 120) GO TO 70                          LLP08600
      LINE(L)=ISTAR                                                   LLP08700
   70 CONTINUE                                                        LLP08800
      IF (15-I) 90,80,90                                              LLP08900
   80 IEXP=IPWRY+2                                                    LLP09000
      GO TO 130                                                       LLP09100
   90 IF (31-I) 110,100,110                                           LLP09200
  100 IEXP=IPWRY+1                                                    LLP09300
      GO TO 130                                                       LLP09400
  110 IF (47-I) 140,120,140                                           LLP09500
  120 IEXP=IPWRY                                                      LLP09600
  130 IF (I .GT. 1) WRITE (6,2040) IAI,IEXP,LINE                      LLP09700
      GO TO 220                                                       LLP09800
  140 IF (16-I) 150,170,150                                           LLP09900
  150 IF (32-I) 160,170,160                                           LLP10000
  160 IF (48-I) 175,170,175                                           LLP10100
  170 IF (I .EQ. 1) GO TO 220                                         LLP10200
      IF (I .EQ. 48) GO TO 171                                        LLP10300
      WRITE (6,2050) IAI,LINE                                         LLP10400
      GO TO 220                                                       LLP10500
  171 IP1 = IPWRX+1                                                   LLP10600
      IP2 = IPWRX+2                                                   LLP10700
      IP3 = IPWRX+3                                                   LLP10800
      WRITE (6,2020) IPWRX,IP1,IP2,IP3,FLDX                           LLP10900
      GO TO 220                                                       LLP11000
  175 IEXP = 3                                                        LLP11100
      IF (MOD(I,16) .NE. 5) GO TO 180                                 LLP11200
      IEXP = 1                                                        LLP11300
      GO TO 190                                                       LLP11400
  180 IF (MOD(I,16) .EQ. 11) IEXP = 2                                 LLP11500
  190 IF (I .GT. 1) WRITE (6,2030) IAI,LEXP(IEXP),LINE                LLP11600
  220 CONTINUE                                                        LLP11700
      IF (NCRIT .GT. 0.AND.NCRIT .LT. 9) WRITE (6,2080) (LCRIT(I),CRIT(ILLP11800
     1),I=1,NCRIT)                                                    LLP11900
  370 RETURN                                                          LLP12000
  500 WRITE(6,2090)                                                   LLP12100
      RETURN                                                          LLP12200
      END                                                             LLP12300
```

```
      SUBROUTINE LABELS(K)                                              LBL00100
      COMMON /PARAMT/ TESTNO(12),        ISKIP(15),NXS,NYS,NZS,NDI,NCI, LBL00200
     1NBK,NPTS,NVS,NVB,XX(41),YY(41),Z(16),DELX(15),DELY(15),Q(15),     LBL00300
     2UBARK(16),SIGAK(16),SIGEK(16),SIGXO(15),SIGYO(15),SIGZO(15),      LBL00400
     3ALPHA(20),BETA(20),ZRK,TIMAV,THETAK(16),TAUK,TAUOK,H,XRY,XRZ,     LBL00500
     4XLRY,XLRZ,ZZL(40),IZMOD(15),DECAY,ZLIM,TIM1,LAMBDA,DI(10),CI(10), LBL00600
     5TAST(05),JBOT(05),JTOP(05),VS(20),PERC(20),ACCUR,VB(20),PERCB(20),LBL00700
     6HB,ALPHL(05),BETL(05),TAUL,TAUOL,ZRL,UBARL(10),SIGAL(10),SIGEL(10)LBL00800
     7,THETAL(10),GAMMAP(20),NTI,TI(10),NPS,NAMCAS(12)                  LBL00900
      COMMON /PARAMS/ UBAR(20),SIGAP(20),DELTHP(20),SIGEP(20),THETA(20),LBL01000
     1DELU(20),VER,VREF,PEAKD,SIGZ,SIGY,SIGX,SQR2P,L,TH,I,J,KK,STO1,    LBL01100
     2STO2,STO3,TRD,ILK,RAD,NNZ,ITOP,IBOT,XAST(21),SIGXNK,JF,PPWR,QPWR, LBL01200
     3MPWR,II,DEP,XBARX,SQBAR,NXCI,LAT,SIGYNK,GAMMA(20),NCC,NDD,NTT,    LBL01300
     4NCCC,NDDD,NTTT,NSW2,MODLS(15),KSW(5),LINES,IM1,MDLS,NWD,          LBL01400
     5YSV(41),YBARY(41),UBARNK(41),BETANK(41),ALPHNK(41),ANG(42),       LBL01500
     6SIGENK(41),SIGANK(41),DEPN(41,41),RNG,AZM,IDATE(2),ITIME(2),YT,   LBL01600
     7NYSS,CDAMX(3)                                                     LBL01700
      DIMENSION LINE(1)                                                 LBL01800
      COMMON /LBLLBL/ J1(9),J2(4),J3(48),J5(6),J7(3),J8(16),J9(13),J10, LBL01900
     1J4(12),J11(2),UNIT(15)                                            LBL02000
      EQUIVALENCE (YBARY,LINE)                                          LBL02100
      INTEGER TESTNO                                                    LBL02200
      DATA IBLNK/6H      /                                              LBL02300
C     CHANGE FOLLOWING TWO STATEMENTS FOR 7044                          LBL02400
C     DATA IBLK/000000000060/,IBLP/000000000033/                        LBL02500
      DATA IBLK/000000000005/,IBLP/000000000075/                        LBL02600
      DATA J1/54HCALCULATIONS OF   MAXIMUM CENTERLINEISOPLETHS          LBL02700
      DATA J11/7H MINUTE/                                               LBL02800
      DATA J2/24H HCL    CO     CO2   AL2O3/                            LBL02900
      DATA J3/14H CONCENTRATION,3*1H ,7H DOSAGE,4*1H ,24H TIME-MEAN CONCLBL03000
     1ENTRATION,2*2H  ,25H GRAVITATIONAL DEPOSITION,1H ,13H CALCULATIONSLBL03100
     2,3*1H ,25H PRECIPITATION DEPOSITION,1H ,22H TIME OF CLOUD PASSAGE,LBL03200
     32*1H ,28H AVERAGE CLOUD CONCENTRATION,1H /                        LBL03300
      DATA J10/3H IN/                                                   LBL03400
      DATA J4/3HPPM,1H ,6HPPM SE,1HC,6HMG/M**,1H3,6HMG SEC,5H/M**3,6HMG/LBL03500
     1M**,1H2,6HSECOND,1HS/                                             LBL03600
      DATA J5/36H AT A HEIGHT OF                METERS/                 LBL03700
      DATA J7/18H DOWNWIND FROM A  /                                    LBL03800
      DATA J8/96H STATIC FIRE.          NORMAL LAUNCH.          SINGLELBL03900
     1 ENGINE BURN,     SLOW BURN.          /                           LBL04000
```

```
      DATA J9/78HMODEL         WAS USED IN THE CALCULATIONS AND THE METEORLBL04100
     1OLOGICAL CASE IS                                                     LBL04200
      DO 10 J=1,70                                                         LBL04300
   10 LINE(J) = IBLNK                                                      LBL04400
      DO 20 J=1,3                                                          LBL04500
      N = KSW(3)+J                                                         LBL04600
   20 LINE(J+3) = J1(N)                                                    LBL04700
   30 IF (TESTNO(11) .NE. IBLNK) GO TO 40                                  LBL04800
      N = ISKIP(5)                                                         LBL04900
      LINE(8) = J2(N)                                                      LBL05000
      GO TO 50                                                             LBL05100
   40 LINE(8) = TESTNO(11)                                                 LBL05200
      LINE(9) = TESTNO(12)                                                 LBL05300
   50 IF (KSW(4) .NE. 12) GO TO 60                                         LBL05400
      B = TIMAV/60.0                                                       LBL05500
      CALL NMBRS(B,LINE(10),IDUM)                                          LBL05600
      LINE(12) = J11(1)                                                    LBL05700
      LINE(13) = J11(2)                                                    LBL05800
   60 DO 70 J=1,6                                                          LBL05900
      N = KSW(4)+J                                                         LBL06000
   70 LINE(J+13) = J3(N)                                                   LBL06100
      IF (KSW(4) .NE. 24) LINE(20) = J10                                   LBL06200
      IF (KSW(4) .EQ. 24) GO TO 100                                        LBL06300
      M = 8                                                                LBL06400
      IF (KSW(4) .EQ. 30.OR.KSW(4) .EQ. 18) GO TO 80                       LBL06500
      M = 0                                                                LBL06600
      IF (ISKIP(5) .EQ. 4) M = 4                                           LBL06700
      IF (KSW(4) .EQ. 6) M = M+2                                           LBL06800
      IF (KSW(4) .EQ. 36) M = 10                                           LBL06900
   80 DO 90 J=1,2                                                          LBL07000
      N = M+J                                                              LBL07100
   90 LINE(J+20) = J4(N)                                                   LBL07200
  100 DO 110 J=1,6                                                         LBL07300
  110 LINE(J+22) = J5(J)                                                   LBL07400
      CALL NMBRS(ZZL(K),LINE(26),IDUM)                                     LBL07500
  130 DO 140 J=1,3                                                         LBL07600
  140 LINE(J+28) = J7(J)                                                   LBL07700
      DO 150 J=1,4                                                         LBL07800
  150 LINE(J+31) = TESTNO(J+6)                                             LBL07900
      IF (ISKIP(6) .EQ. 5) GO TO 165                                       LBL08000
      DO 160 J=1,4                                                         LBL08100
```

```
      N = ISKIP(6) *4-4+J                                               LBL08200
  160 LINE(J+35) = J8(N)                                                LBL08300
  165 J = 39                                                            LBL08400
      NWD = 1                                                           LBL08500
      GO TO 190                                                         LBL08600
  166 IF (N .EQ. 6) JS = JS+1                                           LBL08700
      DO 170 J=1,13                                                     LBL08800
  170 LINE(J+JS) = J9(J)                                                LBL08900
      B = MDLS                                                          LBL09000
      CALL NMBRS(B,LINE(JS+2),IDUM)                                     LBL09100
      JS = JS+13                                                        LBL09200
      DO 180 J=1,6                                                      LBL09300
  180 LINE(J+JS) = TESTNO(J)                                            LBL09400
      J = JS+6                                                          LBL09500
      NWD = 2                                                           LBL09600
  190 N = 7                                                             LBL09700
  191 N = N-1                                                           LBL09800
      IF (N .GT. 0) GO TO 200                                           LBL09900
      N = 6                                                             LBL10000
      J = J-1                                                           LBL10100
  200 JS = IABS(6*(N-1))                                                LBL10200
      M = 0                                                             LBL10300
      CALL MSFLD(JS,6,LINE(J),30,M)                                     LBL10400
      IF (M .EQ. IBLK) GO TO 191                                        LBL10500
      IF (M .EQ. IBLP) GO TO 220                                        LBL10600
      N = N+1                                                           LBL10700
      IF (N .LT. 7) GO TO 210                                           LBL10800
      N = 1                                                             LBL10900
      J = J+1                                                           LBL11000
  210 JS = IABS(6*(N-1))                                                LBL11100
      CALL MSFLD(30,6,IBLP,JS,LINE(J))                                  LBL11200
  220 JS = J                                                            LBL11300
      GO TO (166,230),NWD                                               LBL11400
  230 NWD = JS                                                          LBL11500
      CALL PACKS(LINE,NWD)                                              LBL11600
      RETURN                                                            LBL11700
      END                                                               LBL11800
```

```
      SUBROUTINE MAXMIN(A,B,C,D,E,F,ISW)                               MMN00100
C     THIS SUBROUTINE DETERMINES MAX & MIN FOR PLOTTING FUNCTION VS.   MMN00200
C     DISTANCE A,B ARE INPUT MAX AND MIN, C AND D ARE CALC MAX AND MIN MMN00300
C     DETERMINE MAX AND MIN                                            MMN00400
      IF (ISW .EQ. 2) GO TO 10                                         MMN00500
      IF (A .GT. 0.0) GO TO 80                                         MMN00600
C     LINEAR SCALING                                                   MMN00700
      C = E                                                            MMN00800
      D = F                                                            MMN00900
      GO TO 90                                                         MMN01000
   10 CONTINUE                                                         MMN01100
      XX = 4.0                                                         MMN01200
      IF (A .GT. 0.0) XX = A                                           MMN01300
C     LOG-LOG SCALING                                                  MMN01400
      C = ALOG10(E)                                                    MMN01500
      K = C                                                            MMN01600
      X = K                                                            MMN01700
      IF (X-C) 20,30,20                                                MMN01800
   20 K = K+1                                                          MMN01900
   30 C = 10.0**K                                                      MMN02000
      D = 1.0                                                          MMN02100
      IF (F .LE. 0.0) GO TO 40                                         MMN02200
      D = F                                                            MMN02300
   40 D = ALOG10(D)                                                    MMN02400
      J = D                                                            MMN02500
      X = J                                                            MMN02600
      IF (X-D) 50,60,50                                                MMN02700
   50 IF (D .LT. 0.0) J = J-1                                          MMN02800
   60 IF (FLOAT(K-J) .LE. XX) GO TO 70                                 MMN02900
      J = J+1                                                          MMN03000
      GO TO 60                                                         MMN03100
   70 D = 10.0**J                                                      MMN03200
      GO TO 90                                                         MMN03300
   80 C = A                                                            MMN03400
      D = B                                                            MMN03500
   90 RETURN                                                           MMN03600
      END                                                              MMN03700
```

```
      SUBROUTINE LSSOPT(X,Y,NX,NY,FI,VLABEL,LEGEND,NCV,NCHAR)           LSS00100
C     IS ISW = 2 LOG-LOG, IF ISW = 1 LINEAR                             LSS00200
      COMMON /PLTLLO/ ISW,XMAXJN,YMAXJN,XCIZE,YCIZE                     LSS00300
      COMMON /BNDS/ XRIT,XLFT,YBOT,YTOP,XPL,YPL                         LSS00400
      DIMENSION X(1),Y(1),FI(1)                                         LSS00500
      COMMON /ILALPH/LCRIT(10),IBLANK,ISTAR,IP1,IP2,IP3,HLABEL(5),NCH   LSS00600
      COMMON /XYXYPT/ YP(41),XP(41),A(41),B(41),C(41),D(41),XI(41),YI(41LSS00700
     1),NUM(3),NC                                                       LSS00800
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, LSS00900
     1XSIZE1,YSIZE1                                                     LSS01000
      DATA DISP,XLN,YBN,XRN,YTN/3.,62.,102.,24.,22./                    LSS01100
      XLM1 = XLN                                                        LSS01200
      YBM1 = YBN                                                        LSS01300
      XRM1 = XRN                                                        LSS01400
      YTM1 = YTN                                                        LSS01500
      XSIZE1 = XCIZE                                                    LSS01600
      YSIZE1 = YCIZE                                                    LSS01700
C     DETERMINE MAX AND MIN FOR BOTH AXES                               LSS01800
      Y1 = 0.0                                                          LSS01900
      J1 = 0                                                            LSS02000
      J2 = 0                                                            LSS02100
      Y2 = 1.0E20                                                       LSS02200
      J3 = 0                                                            LSS02300
      DO 50 I=1,NX                                                      LSS02400
      IF (Y(I) .LE. 0.0) GO TO 40                                       LSS02500
      IF (J1 .GT. 0) GO TO 30                                           LSS02600
      J1 = I                                                            LSS02700
      GO TO 31                                                          LSS02800
   30 J2 = I                                                            LSS02900
   31 IF (Y(I) .GT. Y1) Y1 = Y(I)                                       LSS03000
      IF (Y(I) .LT. Y2) Y2 = Y(I)                                       LSS03100
      GO TO 50                                                          LSS03200
   40 J3 = 1                                                            LSS03300
   50 CONTINUE                                                          LSS03400
      IF (ISW .EQ. 2) GO TO 60                                          LSS03500
      IF (J3 .EQ. 1) Y2 = 0.0                                           LSS03600
      J1 = 1                                                            LSS03700
      J2 = NX                                                           LSS03800
      CALL MAXMIN(XMAXJN,X(J1),XMAX,XMIN,X(J2),X(J1),1)                 LSS03900
      CALL MAXMIN(YMAXJN,Y2,YMAX,YMIN,Y1,Y2,1)                          LSS04000
```

```
      GO TO 80                                                          LSS04100
   60 IF (Y1 .LE. 0.0) GO TO 230                                        LSS04200
   61 IF (X(J1) .GT. 0.0) GO TO 70                                      LSS04300
      J1 = J1+1                                                         LSS04400
      GO TO 61                                                          LSS04500
   70 XPL = 10.0**X(J1)                                                 LSS04600
      YPL = 10.0**X(J2)                                                 LSS04700
      CALL MAXMIN(XMAXJN,XPL,XMAX,XMIN,YPL,XPL,2)                       LSS04800
      CALL MAXMIN(YMAXJN,Y2,YMAX,YMIN,Y1,Y2,2)                          LSS04900
   80 CONTINUE                                                          LSS05000
C     DETERMINE PLOT SCALE                                              LSS05100
   90 IF (ISW .EQ. 2) GO TO 100                                         LSS05200
      SCLX = XSIZE1/(XMAX-XMIN)                                         LSS05300
      SCLY = YSIZE1/(YMAX-YMIN)                                         LSS05400
      GO TO 110                                                         LSS05500
  100 SCLX = XSIZE1/(ALOG10(XMAX)-ALOG10(XMIN))                         LSS05600
      SCLY = YSIZE1/(ALOG10(YMAX)-ALOG10(YMIN))                         LSS05700
  110 CONTINUE                                                          LSS05800
      IF (ISW .EQ. 2) GO TO 115                                         LSS05900
      XLFT = XMIN                                                       LSS06000
      XRIT = XMAX                                                       LSS06100
      YTOP = YMAX                                                       LSS06200
      YBOT = YMIN                                                       LSS06300
      GO TO 116                                                         LSS06400
  115 CONTINUE                                                          LSS06500
      XLFT = ALOG10(XMIN)                                               LSS06600
      XRIT = ALOG10(XMAX)                                               LSS06700
      YTOP = ALOG10(YMAX)                                               LSS06800
      YBOT = ALOG10(YMIN)                                               LSS06900
  116 CONTINUE                                                          LSS07000
      CALL SETMIV(0,0,0,0)                                              LSS07100
      CALL FRAMEV(0)                                                    LSS07200
      HT = 12                                                           LSS07300
      CHARF = 8                                                         LSS07400
C                                                                       LSS07500
C     DRAW AXES                                                         LSS07600
      CALL ILAXES(ISW,VLABEL,HLABEL,NCV,NCH,LEGEND,NCHAR)               LSS07700
C                                                                       LSS07800
C     PLOT CURVE                                                        LSS07900
      NC = J2-J1+1                                                      LSS08000
      IF (NC .LE. 0) GO TO 220                                          LSS08100
```

```
      IF (ISW .NE. 2) GO TO 121
      DO 120 I=J1,J2
      Y(I) = ALOG10(Y(I))
  120 CONTINUE
  121 CONTINUE
      IF (NC .LT. 3) GO TO 125
      CALL SPLINE(X(J1),Y(J1),A,B,C,D,NC,IER)
      IF (IER .EQ. 1) GO TO 125
      DX = (X(J2)-X(J1))/82.0
      XPL = X(J1)-DX
      N = 0
      I = 1
  123 XPL = XPL+DX
      IF (XPL .LT. X(J1+I)) GO TO 124
      I = I+1
  124 IF (I+J1 .GT. J2.OR.N .GE. 82) GO TO 127
      N = N+1
      YPL = XPL-X(J1+I-1)
      YP(N) = Y(J1+I-1)+YPL*(B(I)+(XPL-X(J1+I))*(2.0*C(I)+C(I+1)+A(I)*
     1YPL)*.1666667)
      XI(N) = XPL
      GO TO 123
  125 DO 126 I=1,NC
      XI(I) = X(J1+I-1)
  126 YP(I) = Y(J1+I-1)
      N = NC
  127 CONTINUE
      CALL ILPLOT(XI,YP,N,2,A)
      IF (NY .EQ. 0.OR.NY .EQ. 9) GO TO 220
      XINC = 8
      IF (ISW .EQ. 2) YMIN = ALOG10(YMIN)
      DO 210 I=1,NY
      IF (ISW .EQ. 2) GO TO 130
      YS = (FI(I)-YMIN)*SCLY+YBM1
      GO TO 131
  130 YS = (ALOG10(FI(I))-YMIN)*SCLY+YBM1
  131 IF (YS .LT. YBM1) GO TO 210
      IF (YS .GT. YBM1+YSIZE1) GO TO 210
      CALL NMBRS(FI(I),NUM,NC)
      X1 = XLM1-XINC
      X2 = XLM1+XSIZE1-FLOAT(NC)*CHARF-DISP
```

```
      IB = 2
      XPL = XLM1
  140 X1 = X1+XINC
      IF (X1 .GE. X2) X1 = X2
      IF (IB .EQ. 3) GO TO 150
      CALL LINE2V(IFIX(XPL),IFIX(YS),IFIX(X1-XPL),0)
  150 IF (X1 .GE. X2) GO TO 170
      XPL = X1
      IF (IB .EQ. 2) GO TO 160
      IB = 2
      GO TO 140
  160 IB = 3
      GO TO 140
  170 CALL PRINTV(NC,NUM,IFIX(X2+XINC),IFIX(YS-4.0))
  210 CONTINUE
  220 CONTINUE
  230 RETURN
      END
```

```
      SUBROUTINE FSTPLT(A,B,C,I,J,K,D,E,F,V1,V2,V3,N1,N2,N3,XP)        FST00100
      DIMENSION I(1),J(1),K(1),LSTK(2),V1(1),V2(1),V3(1)               FST00200
      DATA MTRS/6HMETERS/                                              FST00300
      IF (K(1) .EQ. LSTK(1).AND.K(2) .EQ. LSTK(2)) GO TO 40           FST00400
      CALL SETMIV(0,0,0,0)                                             FST00500
      CALL FRAMEV(0)                                                   FST00600
      CALL PRINTV(72,I,200,800)                                        FST00700
      CALL PRINTV(37,37HADJUSTED CLOUD STABILIZATION HEIGHT =,200,750) FST00800
      CALL LABLV(A,504,750,7,1,4)                                      FST00900
      CALL PRINTV(6,MTRS,568,750)                                      FST01000
      CALL PRINTV(7,7HRANGE =,200,725)                                 FST01100
      CALL LABLV(B,272,725,7,1,5)                                      FST01200
      CALL PRINTV(6,MTRS,336,725)                                      FST01300
      CALL PRINTV(17,17HAZIMUTH BEARING =,200,700)                     FST01400
      CALL LABLV(C,352,700,6,1,3)                                      FST01500
      CALL PRINTV(7,7HDEGREES,416,700)                                 FST01600
      CALL PRINTV(8,8HRUN DATE,200,650)                                FST01700
      CALL PRINTV(8,J,288,650)                                         FST01800
      CALL PRINTV(8,8HRUN TIME,400,650)                                FST01900
      CALL PRINTV(8,K,488,650)                                         FST02000
      IDY = 625                                                        FST02100
      IF (N1 .LT. 0) GO TO 10                                          FST02200
      IDY = IDY-25                                                     FST02300
      CALL PRINTV(7,7HMAXIMUM,200,IDY)                                 FST02400
      CALL PRINTV(N1,V1,264,IDY)                                       FST02500
      CALL LABLV(D,264+8*(N1+1),IDY,-6,1,0)                            FST02600
   10 IF (N2 .LT. 0) GO TO 20                                          FST02700
      IDY = IDY-25                                                     FST02800
      CALL PRINTV(7,7HMAXIMUM,200,IDY)                                 FST02900
      CALL PRINTV(N2,V2,264,IDY)                                       FST03000
      CALL LABLV(E,264+8*(N2+1),IDY,-6,1,0)                            FST03100
   20 IF (N3 .LT. 0) GO TO 30                                          FST03200
      IDY = IDY-25                                                     FST03300
      CALL PRINTV(7,7HMAXIMUM,200,IDY)                                 FST03400
      CALL LABLV(XP,264,IDY,4,1,2)                                     FST03500
      CALL PRINTV(6,6HMINUTE,304,IDY)                                  FST03600
      CALL PRINTV(N3,V3,352,IDY)                                       FST03700
      CALL LABLV(F,352+8*(N3+1),IDY,-6,1,0)                            FST03800
   30 CONTINUE                                                         FST03900
      LSTK(1) = K(1)                                                   FST04000
```

```
      LSTK(2) = K(2)                                                    FST04100
   40 RETURN                                                            FST04200
      END                                                               FST04300
```

```
      SUBROUTINE NMBRS(A,NUM,NC)                                        NMB00100
      DIMENSION IM(15),NUM(3)                                           NMB00200
      NC = 0                                                            NMB00300
      IF (A) 20,10,30                                                   NMB00400
   10 NC = 1                                                            NMB00500
      NUM(1) = '0                                                       NMB00600
      GO TO 110                                                         NMB00700
   20 NC = NC+1                                                         NMB00800
      IM(1) = '     -                                                   NMB00900
   30 B = ABS(A)                                                        NMB01000
      K = 0                                                             NMB01100
      M = B                                                             NMB01200
      IF (M .EQ. 0) GO TO 41                                            NMB01300
      M = ALOG10(B)                                                     NMB01400
      M = M+1                                                           NMB01500
      MM = B                                                            NMB01600
      DO 40 I=1,M                                                       NMB01700
      NC = NC+1                                                         NMB01800
      K = MM/10**(M-I)                                                  NMB01900
      MM = MM-K*10**(M-I)                                               NMB02000
   40 IM(NC) = K+48                                                     NMB02100
      K = 3                                                             NMB02200
   41 M = B                                                             NMB02300
      C = M                                                             NMB02400
      IF (B-C) 50,80,50                                                 NMB02500
   50 NC = NC+1                                                         NMB02600
      IM(NC) =                                                          NMB02700
      B = B-M                                                           NMB02800
      I = 0                                                             NMB02900
      B = B+1.0E-7                                                      NMB03000
   60 I = I+1                                                           NMB03100
      NC = NC+1                                                         NMB03200
      B = B*10.0                                                        NMB03300
      M = B                                                             NMB03400
      B = B-M                                                           NMB03500
      IM(NC) = M+48                                                     NMB03600
      IF (I .LT. 6) GO TO 60                                            NMB03700
   70 IF (IM(NC) .GT. 48.AND.IM(NC) .LT. 58) GO TO 80                   NMB03800
      NC = NC-1                                                         NMB03900
      IF (NC .LE. 2) GO TO 80                                           NMB04000
```

```
      GO TO 70                                                          NMB04100
   80 K = 1                                                             NMB04200
      M = 0                                                             NMB04300
      DO 100 I=1,NC                                                     NMB04400
      M = M+1                                                           NMB04500
      IF (M .LT. 7) GO TO 90                                            NMB04600
      M = 1                                                             NMB04700
      K = K+1                                                           NMB04800
   90 CALL MSFLD(30,6,IM(I),IABS(6*(M-1)),NUM(K))                       NMB04900
  100 CONTINUE                                                          NMB05000
  110 CONTINUE                                                          NMB05100
      RETURN                                                            NMB05200
      END                                                               NMB05300
```

```
      SUBROUTINE ILAXES(ISW,VLABEL,HLABEL,NCV,NCH,LEGEND,NCHAR)         ILA00100
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, ILA00200
     1XSIZE1,YSIZE1                                                     ILA00300
      DIMENSION NUM(3),LEGEND(1),VLABEL(1),HLABEL(1)                    ILA00400
C     ISW = 1 LINEAR AXES                                               ILA00500
C     ISW = 2 LOG-LOG AXES                                              ILA00600
      DATA TIC1/8.0/,TIC2/4.0/,DISP/3.0/                                ILA00700
      IF (ISW .NE. 2) GO TO 40                                          ILA00800
      XST = ALOG10(XMIN)                                                ILA00900
      XINC = 1.0                                                        ILA01000
      K = XST                                                           ILA01100
      XP = K                                                            ILA01200
      IF (XST-XP) 20,60,20                                              ILA01300
   20 IF (XST) 21,21,30                                                 ILA01400
   21 K = K-1                                                           ILA01500
      GO TO 60                                                          ILA01600
   30 K = K+1                                                           ILA01700
      GO TO 60                                                          ILA01800
   40 CONTINUE                                                          ILA01900
C     DETERMINE INCREMENT BETWEEN MINOR TIC MARKS                       ILA02000
      XINC = (XMAX-XMIN)/(XSIZE1*10.0)                                  ILA02100
      IF (XINC .LT. 1.0) XINC = 1.0                                     ILA02200
      J = ALOG10(XINC)                                                  ILA02300
      K = XINC*10.0**(-J)                                               ILA02400
      XINC = K*10**J                                                    ILA02500
      XST = 0.0                                                         ILA02600
   50 IF (XST .LE. XMIN) GO TO 60                                       ILA02700
      XST = XST-10.0*XINC                                               ILA02800
      GO TO 50                                                          ILA02900
   60 CONTINUE                                                          ILA03000
      CALL LINE2V(IFIX(XLM1),IFIX(YBM1),0,IFIX(YSIZE1))                 ILA03100
      CALL LINE2V(IFIX(XLM1),IFIX(YBM1+YSIZE1),IFIX(XSIZE1),0)          ILA03200
      CALL LINE2V(IFIX(XLM1+XSIZE1),IFIX(YBM1+YSIZE1),0,-IFIX(YSIZE1))  ILA03300
      CALL LINE2V(IFIX(XLM1+XSIZE1),IFIX(YBM1),-IFIX(XSIZE1),0)         ILA03400
C     PLOT AND LABEL X AXES                                             ILA03500
      YP = YBM1                                                         ILA03600
      J = 1                                                             ILA03700
      DO 150 I=1,2                                                      ILA03800
      IF (ISW .EQ. 2) GO TO 70                                          ILA03900
      X = XST-XINC                                                      ILA04000
```

```
      GO TO 80                                                          ILA04100
   70 L = K -1                                                          ILA04200
      X = 9.0                                                           ILA04300
   80 IB = 9                                                            ILA04400
   90 X = X+XINC                                                        ILA04500
      IB = IB+1                                                         ILA04600
      IF (IB .LE. 10) GO TO 95                                          ILA04700
      IB = 1                                                            ILA04800
      IF (ISW .NE. 2) GO TO 95                                          ILA04900
      IB = 2                                                            ILA05000
      L = L+1                                                           ILA05100
      X = 2.0                                                           ILA05200
   95 CONTINUE                                                          ILA05300
      IF (ISW .EQ. 2) GO TO 100                                         ILA05400
      XP = (X-XMIN)*SCLX+XLM1                                           ILA05500
      GO TO 105                                                         ILA05600
  100 XP = (ALOG10(X*10.0**L)-XST)*SCLX+XLM1                            ILA05700
  105 IF (XP .LT. XLM1) GO TO 90                                        ILA05800
      IF (XP .GT. XLM1+XSIZE1) GO TO 140                                ILA05900
  110 CONTINUE                                                          ILA06000
      IF (IB .LT. 10) GO TO 130                                         ILA06100
      CALL LINE2V(IFIX(XP),IFIX(YP),0,IFIX(TIC1)*J)                     ILA06200
      IF (ISW .EQ. 2) GO TO 120                                         ILA06300
      IB = 0                                                            ILA06400
      IF (J .LT. 0) GO TO 140                                           ILA06500
      CALL NMBRS(X,NUM,NCHT)                                            ILA06600
      CALL PRINTV(NCHT,NUM,IFIX(XP-.5*FLOAT(NCHT)*CHARF),IFIX(YP-HT-DISPILA06700
     1))                                                                ILA06800
      GO TO 140                                                         ILA06900
  120 X = L+1                                                           ILA07000
      IB = 1                                                            ILA07100
      IF (J .LT. 0) GO TO 125                                           ILA07200
      CALL PRINTV(2,'10',IFIX(XP-2.0*CHARF),IFIX(YP-HT-3.0*DISP))       ILA07300
      CALL LABLV(X,IFIX(XP),IFIX(YP-HT-DISP),2,1,2)                     ILA07400
  125 CONTINUE                                                          ILA07500
      X = 1.0                                                           ILA07600
      L = L+1                                                           ILA07700
      GO TO 140                                                         ILA07800
  130 CALL LINE2V(IFIX(XP),IFIX(YP),0,IFIX(TIC2)*J)                     ILA07900
  140 CONTINUE                                                          ILA08000
      IF (XP .LT. XLM1+XSIZE1) GO TO 90                                 ILA08100
```

```
      YP = YBM1+YSIZE1                                                  ILA08200
      J = -1                                                            ILA08300
  150 CONTINUE                                                          ILA08400
C     PLOT AND LABEL Y AXES                                             ILA08500
      IF (ISW .NE. 2) GO TO 154                                         ILA08600
      XST = ALOG10(YMIN)                                                ILA08700
      XINC = 1.0                                                        ILA08800
      K = XST                                                           ILA08900
      XP = K                                                            ILA09000
      IF (XST-XP) 151,156,151                                           ILA09100
  151 IF (XST) 152,152,153                                              ILA09200
  152 K = K-1                                                           ILA09300
      GO TO 156                                                         ILA09400
  153 K = K+1                                                           ILA09500
      GO TO 156                                                         ILA09600
  154 XINC = (YMAX-YMIN)/(YSIZE1*10.0)                                  ILA09700
      IF (XINC .LT. 1.0) XINC = 1.0                                     ILA09800
      J = ALOG10(XINC)                                                  ILA09900
      K = XINC*10.0**(-J)                                               ILA10000
      XINC = K*10**J                                                    ILA10100
      XST = 0.0                                                         ILA10200
  155 IF (XST .LE. YMIN) GO TO 156                                      ILA10300
      XST = XST-XINC*10.0                                               ILA10400
      GO TO 155                                                         ILA10500
  156 CONTINUE                                                          ILA10600
      XP = XLM1                                                         ILA10700
      XD = 5.0                                                          ILA10800
      IF (ISW .EQ. 2) XD = 4.0                                          ILA10900
      XD = XP-XD*CHARF-DISP                                             ILA11000
      J = 1                                                             ILA11100
      DO 250 I=1,2                                                      ILA11200
      IF (ISW .EQ. 2) GO TO 160                                         ILA11300
      X = XST-XINC                                                      ILA11400
      GO TO 170                                                         ILA11500
  160 L = K-1                                                           ILA11600
      X = 9.0                                                           ILA11700
  170 IB = 9                                                            ILA11800
  180 X = X+XINC                                                        ILA11900
      IB = IB+1                                                         ILA12000
      IF (IB .LE. 10) GO TO 185                                         ILA12100
      IB = 1                                                            ILA12200
```

```
      IF (ISW .NE. 2) GO TO 185                                         ILA12300
      IB = 2                                                            ILA12400
      L = L+1                                                           ILA12500
      X = 2.0                                                           ILA12600
  185 CONTINUE                                                          ILA12700
      IF (ISW .EQ. 2) GO TO 190                                         ILA12800
      YP = (X-YMIN)*SCLY+YBM1                                           ILA12900
      GO TO 200                                                         ILA13000
  190 YP = (ALOG10(X*10.0**L)-XST)*SCLY+YBM1                            ILA13100
  200 IF (YP .LT. YBM1) GO TO 180                                       ILA13200
      IF (YP .GT. YBM1+YSIZE1) GO TO 240                                ILA13300
  210 CONTINUE                                                          ILA13400
      IF (IB .LT. 10) GO TO 230                                         ILA13500
      CALL LINE2V(IFIX(XP),IFIX(YP),IFIX(TIC1)*J,0)                     ILA13600
      IF (ISW .EQ. 2) GO TO 220                                         ILA13700
      IB = 0                                                            ILA13800
      IF (J .LT. 0) GO TO 240                                           ILA13900
      CALL NMBRS(X,NUM,NCHT)                                            ILA14000
      XF = XP-FLOAT(NCHT)*CHARF-DISP                                    ILA14100
      CALL PRINTV(NCHT,NUM,IFIX(XF),IFIX(YP-.5*HT))                     ILA14200
      IF (XF .LT. XD) XD = XF                                           ILA14300
      GO TO 240                                                         ILA14400
  220 X = L+1                                                           ILA14500
      IB = 1                                                            ILA14600
      IF (J .LT. 0) GO TO 225                                           ILA14700
      CALL PRINTV(2,'10',IFIX(XP-4.0*CHARF-DISP),IFIX(YP-.5*HT))        ILA14800
      CALL LABLV(X,IFIX(XP-2.0*CHARF-DISP),IFIX(YP+DISP),2,1,2)         ILA14900
  225 CONTINUE                                                          ILA15000
      X = 1.0                                                           ILA15100
      L = L+1                                                           ILA15200
      GO TO 240                                                         ILA15300
  230 CALL LINE2V(IFIX(XP),IFIX(YP),IFIX(TIC2)*J,0)                     ILA15400
  240 CONTINUE                                                          ILA15500
      IF (YP .LT. YBM1+YSIZE1) GO TO 180                                ILA15600
      XP = XLM1+XSIZE1                                                  ILA15700
      J = -1                                                            ILA15800
  250 CONTINUE                                                          ILA15900
C     DRAW VERTICAL AXIS LABEL                                          ILA16000
      XP = XD-DISP-CHARF                                                ILA16100
      YP = (YSIZE1+FLOAT(NCV)*(HT+DISP))*0.5+YBM1                       ILA16200
      CALL APRNTV(0,-IFIX(HT+DISP),NCV,VLABEL,IFIX(XP),IFIX(YP))        ILA16300
```

```
C     DRAW HORIZONTAL AXIS LABEL                                        ILA16400
      XP = (XSIZE1-FLOAT(NCH)*CHARF)*0.5+XLM1                           ILA16500
      YP = YBM1-2.0*(2.5*DISP+HT)                                       ILA16600
      CALL PRINTV(NCH,HLABEL,IFIX(XP),IFIX(YP))                         ILA16700
C     DRAW LEGEND FOR PLOT                                              ILA16800
      XP = (XSIZE1-90.0*CHARF)*0.5+XLM1                                 ILA16900
      YP = YP-2.0*HT                                                    ILA17000
      J = -89                                                           ILA17100
  260 YP = YP-(HT+DISP)                                                 ILA17200
      J = J+90                                                          ILA17300
      K = J+89                                                          ILA17400
      IF (K .GT. NCHAR) K = NCHAR                                       ILA17500
      I = (J/6)+1                                                       ILA17600
      CALL PRINTV(K-J+1,LEGEND(I),IFIX(XP),IFIX(YP))                    ILA17700
      IF (K .LT. NCHAR) GO TO 260                                       ILA17800
      RETURN                                                            ILA17900
      END                                                               ILA18000
```

```
      SUBROUTINE SPLINE(X,Y,A,B,C,D,N,IER)                              SPL00100
      DIMENSION X(1),Y(1),A(1),B(1),C(1),D(1)                           SPL00200
      IER = 0                                                           SPL00300
      C(1) = 0.0                                                        SPL00400
      C(N) = 0.0                                                        SPL00500
      Q = 1.07179677                                                    SPL00600
C     Q = 4.0*(2.0-SQRT(3.0))                                           SPL00700
      NP = N-1                                                          SPL00800
      DO 10 I=1,NP                                                      SPL00900
      A(I) = X(I+1)-X(I)                                                SPL01000
      B(I) = (Y(I+1)-Y(I))/A(I)                                         SPL01100
      IF (I .LT. 2) GO TO 10                                            SPL01200
      C(I) = 2.0*(B(I)-B(I-1))/(A(I-1)+A(I))                            SPL01300
      D(I) = C(I)*1.5                                                   SPL01400
C     D(I) = C(I)*3.0/2.0                                               SPL01500
   10 CONTINUE                                                          SPL01600
      NTM = 0                                                           SPL01700
   20 XM = 0.0                                                          SPL01800
      DO 30 I=2,NP                                                      SPL01900
      YP = C(I+1)                                                       SPL02000
      YP = Q*(((YP-C(I-1))/(1.0+A(I)/A(I-1))-YP)*0.5-C(I)+D(I))         SPL02100
      IF (ABS(YP) .GT. XM) XM = ABS(YP)                                 SPL02200
      C(I) = C(I)+YP                                                    SPL02300
   30 CONTINUE                                                          SPL02400
      NTM = NTM+1                                                       SPL02500
      IF (NTM .LT. 80) GO TO 35                                         SPL02600
      IER = 1                                                           SPL02700
      GO TO 36                                                          SPL02800
   35 CONTINUE                                                          SPL02900
      IF (1.0E-3 .LE. XM) GO TO 20                                      SPL03000
   36 CONTINUE                                                          SPL03100
      DO 40 I=1,NP                                                      SPL03200
      A(I) = (C(I+1)-C(I))/A(I)                                         SPL03300
   40 CONTINUE                                                          SPL03400
      RETURN                                                            SPL03500
      END                                                               SPL03600
```

```
      SUBROUTINE ISSOPT(X,Y,NX,NY,FI,LEGEND,NCHAR,DP,NYS,XX,DR,II,IREC,YISS00100
     1T,ISW3,KLINE,NCV,JM,DECAY,LAMBDA)                                 ISS00200
      COMMON /PLTISO/ SCL,XMAXIN,YMAXIN,XSIZE,YSIZE,RASTIN,JSW          ISS00300
      COMMON /BNDS/ XRIT,XLFT,YBOT,YTOP,XPL,YPL                         ISS00400
      DIMENSION X(1),Y(1),FI(1),DP(41,1),XX(1),DR(245,1),NPP(4),YY(1)   ISS00500
      DIMENSION TLABEL(3)                                               ISS00600
      DIMENSION LEGEND(1),KLINE(1)                                      ISS00700
      REAL LAMBDA                                                       ISS00800
      DIMENSION KB(4)                                                   ISS00900
      COMMON /XYXYPT/ YP(41),XP(41),A(41),B(41),C(41),D(41),XI(41),YI(41ISS01000
     1),NUM(3),NC                                                       ISS01100
      EQUIVALENCE (YP,YY)                                               ISS01200
      DATA XLM,YBM,XRM,YTM,DISP/62.,102.,24.,22.,3./                    ISS01300
      DATA RADI/57.295779/                                              ISS01400
      DATA TLABEL/17HDISTANCE (METERS)/,NCTH/17/                        ISS01500
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, ISS01600
     1XSIZE1,YSIZE1                                                     ISS01700
      COMMON /ILALPH/ LCRIT(10),IBLANK,ISTAR,IP1,IP2,IP3,HLABEL(5),NCH  ISS01800
      DATA RAD/.017453293/                                              ISS01900
      XYTERP(A,B,C,D,E) = B-(D-E)*(B-A)/(D-C)                           ISS02000
      NWD = NCHAP/6                                                     ISS02100
      IF (ISW3 .EQ. 1) GO TO 50                                         ISS02200
      XLM1 = XLM                                                        ISS02300
      YBM1 = YBM                                                        ISS02400
      XRM1 = XRM                                                        ISS02500
      YTM1 = YTM                                                        ISS02600
      XSIZE1 = XSIZE                                                    ISS02700
      YSIZE1 = YSIZE                                                    ISS02800
      ISW = 1                                                           ISS02900
C     DETERMINE MAX AND MIN FOR BOTH AXES                               ISS03000
      DO 10 I=1,NX                                                      ISS03100
      DO 10 J=1,NYS                                                     ISS03200
      DO 10 K=1,NY                                                      ISS03300
      IF (DP(I,J) .GT. FI(K)) GO TO 11                                  ISS03400
   10 CONTINUE                                                          ISS03500
      GO TO 800                                                         ISS03600
   11 CONTINUE                                                          ISS03700
      XMX = X(NX-2)                                                     ISS03800
      IF (XMAXIN .GT. 0.0) XMX = XMAXIN                                 ISS03900
      XMAX = 0.0                                                        ISS04000
```

```
      XMIN = 0.0                                                      ISS04100
      YMAX = 0.0                                                      ISS04200
      YMIN = 0.0                                                      ISS04300
      XI(1) = XMX*SIN(YT*RAD)                                         ISS04400
      YI(1) = XMX*COS(YT*RAD)                                         ISS04500
      YPL = 1.0E8                                                     ISS04600
      DO 12 N=1,NY                                                    ISS04700
   12 YPL = AMIN1(YPL,FI(N))                                          ISS04800
      I1 = 0                                                          ISS04900
      I2 = 0                                                          ISS05000
      DO 15 J=1,NYS                                                   ISS05100
      DO 14 I=1,NX                                                    ISS05200
      IF (DP(I,J) .LT. YPL) GO TO 14                                  ISS05300
      IF (I1 .GT. 0) GO TO 13                                         ISS05400
      I1 = I                                                          ISS05500
   13 I2 = I                                                          ISS05600
   14 CONTINUE                                                        ISS05700
      IF (I1 .GT. 0) GO TO 16                                         ISS05800
   15 CONTINUE                                                        ISS05900
   16 J1 = J                                                          ISS06000
      I3 = 0                                                          ISS06100
      I4 = 0                                                          ISS06200
      DO 19 J=1,NYS                                                   ISS06300
      DO 18 I=1,NX                                                    ISS06400
      IF (DP(I,NYS-J+1) .LT. YPL) GO TO 18                            ISS06500
      IF (I3 .GT. 0) GO TO 17                                         ISS06600
      I3 = I                                                          ISS06700
   17 I4 = I                                                          ISS06800
   18 CONTINUE                                                        ISS06900
      IF (I3 .GT. 0) GO TO 20                                         ISS07000
   19 CONTINUE                                                        ISS07100
   20 J2 = NYS-J+1                                                    ISS07200
      IF (I1 .EQ. 0) I1 = 1                                           ISS07300
      IF (I2 .EQ. 0) I2 = NX-2                                        ISS07400
      IF (I3 .EQ. 0) I3 = 1                                           ISS07500
      IF (I4 .EQ. 0) I4 = NX-2                                        ISS07600
      IF (I2 .GT. NX-2) I2 = NX-2                                     ISS07700
      IF (I4 .GT. NX-2) I4 = NX-2                                     ISS07800
      Y1 = Y(J1)                                                      ISS07900
      Y2 = Y(J2)                                                      ISS08000
      IF (YMAXIN .LE. 0.0) GO TO 21                                   ISS08100
```

```
      XPL = 1.0-YMAXIN/(2.0*XMX*XMX)                                    ISS08200
      IF (XPL .GT. 1.0) XPL = 1.0                                       ISS08300
      IF (XPL .LT. -1.0) XPL = -1.0                                     ISS08400
      XPL = ACOS(XPL)*RADI                                              ISS08500
      Y1 = YT+0.5*XPL                                                   ISS08600
      Y2 = YT-0.5*XPL                                                   ISS08700
   21 XI(2) = X(I1)*SIN(Y1*RAD)                                         ISS08800
      YI(2) = X(I1)*COS(Y1*RAD)                                         ISS08900
      XI(3) = X(I3)*SIN(Y2*RAD)                                         ISS09000
      YI(3) = X(I3)*COS(Y2*RAD)                                         ISS09100
      XI(4) = X(I2)*SIN(Y1*RAD)                                         ISS09200
      YI(4) = X(I2)*COS(Y1*RAD)                                         ISS09300
      XI(5) = X(I4)*SIN(Y2*RAD)                                         ISS09400
      YI(5) = X(I4)*COS(Y2*RAD)                                         ISS09500
      DO 22 I=1,5                                                       ISS09600
      XMAX = AMAX1(XMAX,XI(I))                                          ISS09700
      YMAX = AMAX1(YMAX,YI(I))                                          ISS09800
      XMIN = AMIN1(XMIN,XI(I))                                          ISS09900
   22 YMIN = AMIN1(YMIN,YI(I))                                          ISS10000
C     DETERMINE PLOT SCALE                                              ISS10100
      IF (SCL .LE. 0.0) GO TO 30                                        ISS10200
      SCLX = 12.0/(SCL*.3048)                                           ISS10300
      SCLX = SCLX*RASTIN                                                ISS10400
      SCLY = SCLX                                                       ISS10500
      XPL = (XMAX-XMIN)*SCLX                                            ISS10600
      IF (XPL .LE. XSIZE1) GO TO 24                                     ISS10700
      YPL = XPL-XSIZE1                                                  ISS10800
      XMAX = XMAX-0.5*YPL                                               ISS10900
      XMIN = XMIN+0.5*YPL                                               ISS11000
      IF (XMAX .GE. 0.0) GO TO 23                                       ISS11100
      XMIN = XMIN-XMAX                                                  ISS11200
      XMAX = 0.0                                                        ISS11300
      GO TO 24                                                          ISS11400
   23 IF (XMIN .LE. 0.0) GO TO 24                                       ISS11500
      XMAX = XMAX-XMIN                                                  ISS11600
      XMIN = 0.0                                                        ISS11700
   24 XPL = (YMAX-YMIN)*SCLY                                            ISS11800
      IF (XPL .LE. YSIZE1) GO TO 26                                     ISS11900
      YPL = XPL-YSIZE1                                                  ISS12000
      YMAX = YMAX-0.5*YPL                                               ISS12100
      YMIN = YMIN+0.5*YPL                                               ISS12200
```

```
      IF (YMAX .GE. 0.0) GO TO 25                                       ISS12300
      YMIN = YMIN-YMAX                                                  ISS12400
      YMAX = 0.0                                                        ISS12500
      GO TO 26                                                          ISS12600
   25 IF (YMIN .LE. 0.0) GO TO 26                                       ISS12700
      YMAX = YMAX-YMIN                                                  ISS12800
      YMIN = 0.0                                                        ISS12900
   26 CONTINUE                                                          ISS13000
      GO TO 40                                                          ISS13100
   30 CONTINUE                                                          ISS13200
      SCLX = XSIZE1/(XMAX-XMIN)                                         ISS13300
      SCLY = YSIZE1/(YMAX-YMIN)                                         ISS13400
   40 CONTINUE                                                          ISS13500
      HT = 12                                                           ISS13600
      CHARF = 8                                                         ISS13700
      CALL SETMIV(0,0,0,0)                                              ISS13800
      CALL FRAMEV(0)                                                    ISS13900
C     DRAW AXES                                                         ISS14000
      CALL ILAXES(1,TLABEL,TLABEL,NCTH,NCTH,LEGEND,NCHAR)               ISS14100
      XRIT = XMAX                                                       ISS14200
      XLFT = XMIN                                                       ISS14300
      YBOT = YMIN                                                       ISS14400
      YTOP = YMAX                                                       ISS14500
   50 CONTINUE                                                          ISS14600
      LINES = 57                                                        ISS14700
      DO 710 N=1,NY                                                     ISS14800
      IF (N .EQ. 1) GO TO 240                                           ISS14900
      DO 230 I=1,NX                                                     ISS15000
      KOUT = 4*I-4+IREC                                                 ISS15100
      CALL INTOUT(DP,KOUT,NYS,1,41,I)                                   ISS15200
  230 CONTINUE                                                          ISS15300
  240 CONTINUE                                                          ISS15400
      CALL NMBRS(FI(N),NUM,NC)                                          ISS15500
      DO 400 I=1,4                                                      ISS15600
  400 NPP(I) = 0                                                        ISS15700
      NP = 3                                                            ISS15800
      L = 0                                                             ISS15900
      K = 1                                                             ISS16000
      DO 470 I=1,NX                                                     ISS16100
      JB = 0                                                            ISS16200
      DO 430 J=2,NYS                                                    ISS16300
```

```
      IF (DP(I,J-1) .LE. FI(N).AND.FI(N) .LE. DP(I,J)) GO TO 410          ISS16400
      IF (DP(I,J-1) .GE. FI(N).AND.FI(N) .GE. DP(I,J)) GO TO 410          ISS16500
      GO TO 430                                                           ISS16600
  410 JB = 1                                                              ISS16700
      L = L+1                                                             ISS16800
      NP = NP+1                                                           ISS16900
      Y2 = Y(J)                                                           ISS17000
      IF (ABS(Y(J-1)-Y2) .LT. 180.0) GO TO 420                            ISS17100
      Y2 = 360.0-ABS(Y(J-1)-Y2)+Y(J-1)                                    ISS17200
  420 YY(NP) = XYTERP(Y(J-1),Y2,DP(I,J-1),DP(I,J),FI(N))                  ISS17300
      XX(NP) = X(I)                                                       ISS17400
      IF (NP .GE. 245) GO TO 475                                          ISS17500
  430 CONTINUE                                                            ISS17600
      IF (JB .EQ. 1) GO TO 440                                            ISS17700
      IF (L .EQ. 0) GO TO 440                                             ISS17800
      NPP(K) = L                                                          ISS17900
      L = 0                                                               ISS18000
      K = K+1                                                             ISS18100
  440 IF (I .EQ. NX) GO TO 470                                            ISS18200
      DO 460 J=1,NYS                                                      ISS18300
      IF (DP(I,J) .LE. FI(N).AND.FI(N) .LE. DP(I+1,J)) GO TO 450          ISS18400
      IF (DP(I,J) .GE. FI(N).AND.FI(N) .GE. DP(I+1,J)) GO TO 450          ISS18500
      GO TO 460                                                           ISS18600
  450 L = L+1                                                             ISS18700
      NP = NP+1                                                           ISS18800
      XX(NP) = XYTERP(X(I),X(I+1),DP(I,J),DP(I+1,J),FI(N))                ISS18900
      YY(NP) = Y(J)                                                       ISS19000
      IF (NP .GE. 245) GO TO 475                                          ISS19100
  460 CONTINUE                                                            ISS19200
  470 CONTINUE                                                            ISS19300
  475 CONTINUE                                                            ISS19400
      NP = NP-3                                                           ISS19500
      NPP(K) = L                                                          ISS19600
  480 IF (NPP(K) .GT. 1) GO TO 490                                        ISS19700
      K = K-1                                                             ISS19800
      IF (K .LE. 0) GO TO 710                                             ISS19900
      GO TO 480                                                           ISS20000
  490 CONTINUE                                                            ISS20100
C     DETERMINE IF CLOSED CURVE OR NO, KB(L)=0 IS YES, KB(L) NOT 0 IS NOISS20200
      IP1 = 3                                                             ISS20300
      DO 497 L=1,K                                                        ISS20400
```

```
      KB(L) = 0                                                         ISS20500
      MP = NPP(L)                                                       ISS20600
      J2 = 1                                                            ISS20700
      DO 495 I2=1,2                                                     ISS20800
      DO 491 I=1,MP                                                     ISS20900
      IF (YY(I+IP1)-Y(J2)) 491,492,491                                  ISS21000
  491 CONTINUE                                                          ISS21100
      GO TO 495                                                         ISS21200
  492 X1 = 1.0E8                                                        ISS21300
      DO 493 J=1,MP                                                     ISS21400
      IF (J+IP1 .EQ. I+IP1) GO TO 493                                   ISS21500
      Y1 = ABS(Y(J2)-YY(J+IP1))                                         ISS21600
      IF (Y1 .GT. 180.0) Y1 = 360.0-Y1                                  ISS21700
      IF (Y1 .GE. X1) GO TO 493                                         ISS21800
      X1 = Y1                                                           ISS21900
      J1 = J                                                            ISS22000
  493 CONTINUE                                                          ISS22100
      IF (XX(I+IP1) .GT. XX(J1+IP1)) GO TO 494                          ISS22200
      KB(L) = KB(L)+3                                                   ISS22300
      GO TO 495                                                         ISS22400
  494 KB(L) = KB(L)+1                                                   ISS22500
  495 J2 = NYS                                                          ISS22600
  496 IP1 = IP1+NPP(L)                                                  ISS22700
      IF (KB(L) .EQ. 1.OR.KB(L) .EQ. 3) KB(L) = 0                       ISS22800
  497 CONTINUE                                                          ISS22900
      IF (K .LE. 1) GO TO 501                                           ISS23000
      L = 1                                                             ISS23100
  498 L = L+1                                                           ISS23200
      IF (L .GT. K) GO TO 501                                           ISS23300
      IF (KB(L-1) .NE. 2) GO TO 498                                     ISS23400
      IF (KB(L) .NE. 6) GO TO 498                                       ISS23500
      NPP(L-1) = NPP(L-1)+NPP(L)                                        ISS23600
      KB(L-1) = 0                                                       ISS23700
      MP = 0                                                            ISS23800
  499 IF (L+MP .GE. K) GO TO 500                                        ISS23900
      NPP(L+MP) = NPP(L+MP+1)                                           ISS24000
      KB(L+MP) = KB(L+MP+1)                                             ISS24100
      MP = MP+1                                                         ISS24200
      GO TO 499                                                         ISS24300
  500 K = K-1                                                           ISS24400
      GO TO 498                                                         ISS24500
```

```
  501 CONTINUE                                                     ISS24600
      IP1 = 3                                                      ISS24700
      DO 700 L=1,K                                                 ISS24800
      NP = NPP(L)                                                  ISS24900
C     SHIFT POINTS TO START OF ARRAY                               ISS25000
      I = 0                                                        ISS25100
      DO 502 J=1,NP                                                ISS25200
      I = I+1                                                      ISS25300
      XX(I) = XX(J+IP1)                                            ISS25400
      YY(I) = YY(J+IP1)                                            ISS25500
      IF (XX(I) .LT. X(NX)) GO TO 502                              ISS25600
  502 CONTINUE                                                     ISS25700
      CALL CALCS(XX,YY,NP,RAD,RADI,XSHFT,YSHFT,0.0,0.0,0)          ISS25800
      I = 1                                                        ISS25900
  543 I = I+1                                                      ISS26000
      IF (I .GT. NP) GO TO 546                                     ISS26100
      IF (YY(I)-YY(I-1) .GE. 1.0) GO TO 543                        ISS26200
      I = I+1                                                      ISS26300
      IF (I .GT. NP) GO TO 545                                     ISS26400
      DO 544 J=I,NP                                                ISS26500
      YY(J-1) = YY(J)                                              ISS26600
  544 XX(J-1) = XX(J)                                              ISS26700
      I = I-2                                                      ISS26800
  545 NP = NP-1                                                    ISS26900
  546 CONTINUE                                                     ISS27000
C     FIND START POINT OF CURVE                                    ISS27100
      IF (KB(L) .EQ. 0) GO TO 577                                  ISS27200
C     CALC MAX DIFF AND START INDEX FOR OPEN CURVE                  ISS27300
      YRM = YT                                                     ISS27400
      IF (KB(L) .EQ. 6) YRM = YT-180.0                             ISS27500
  550 IF (YRM .GE. YY(1)) GO TO 551                                ISS27600
      YRM = YRM+360.0                                              ISS27700
      GO TO 550                                                    ISS27800
  551 IF (YRM .LE. YY(NP)) GO TO 552                               ISS27900
      YRM = YRM-360.0                                              ISS28000
      GO TO 551                                                    ISS28100
  552 CONTINUE                                                     ISS28200
      DO 553 I=1,NP                                                ISS28300
      IF (YY(I) .LT. YRM) GO TO 553                                ISS28400
      MP = I-1                                                     ISS28500
  553 CONTINUE                                                     ISS28600
```

```
      GO TO 582                                                         ISS28700
  577 YRM = YT+180.0                                                    ISS28800
  578 IF (YRM .GT. YY(1)) GO TO 579                                     ISS28900
      YRM = YRM+360.0                                                   ISS29000
      GO TO 578                                                         ISS29100
  579 IF (YRM .LE. YY(NP)) GO TO 580                                    ISS29200
      YRM = YRM-360.0                                                   ISS29300
      GO TO 579                                                         ISS29400
  580 MP = 0                                                            ISS29500
      DO 581 I=1,NP                                                     ISS29600
      IF (YY(I) .LT. YRM) GO TO 581                                     ISS29700
      MP = I-1                                                          ISS29800
      GO TO 582                                                         ISS29900
  581 CONTINUE                                                          ISS30000
  582 IF (MP .LE. 0) GO TO 594                                          ISS30100
      DO 592 I=1,MP                                                     ISS30200
      X1 = XX(1)                                                        ISS30300
      Y1 = YY(1)                                                        ISS30400
      DO 591 J=1,NP                                                     ISS30500
      XX(J-1) = XX(J)                                                   ISS30600
  591 YY(J-1) = YY(J)                                                   ISS30700
      XX(NP) = X1                                                       ISS30800
  592 YY(NP) = Y1                                                       ISS30900
C     MAKE SURE IN ASCENDING ORDER                                      ISS31000
      Y1 = YY(1)                                                        ISS31100
      DO 593 I=2,NP                                                     ISS31200
      DIF = ABS(YY(I)-Y1)                                               ISS31300
      IF (DIF .GT. 180.0) DIF = 360.0-DIF                               ISS31400
      Y1 = YY(I)                                                        ISS31500
  593 YY(I) = YY(I-1)+DIF                                               ISS31600
  594 CONTINUE                                                          ISS31700
      IF (NP .GT. 245) NP = 245                                         ISS31800
      DIF = 0.0                                                         ISS31900
      DO 595 I=1,NP                                                     ISS32000
      IF (XX(I) .LE. DIF) GO TO 595                                     ISS32100
      DIF = XX(I)                                                       ISS32200
      N1 = I                                                            ISS32300
  595 CONTINUE                                                          ISS32400
      X1 = YY(N1)                                                       ISS32500
      IF (N1-2 .LT. 1) GO TO 596                                        ISS32600
      IF (N1+2 .GT. NP) GO TO 596                                       ISS32700
```

```
      X1 = (YY(N1-2)+YY(N1-1)+YY(N1)+YY(N1+1)+YY(N1+2))/5              ISS32800
  596 CONTINUE                                                         ISS32900
      NPI = NP                                                         ISS33000
      IER = 1                                                          ISS33100
      IF (NP .LT. 6) GO TO 650                                         ISS33200
      IF (JSW .NE. 0) GO TO 650                                        ISS33300
      CALL SPLINE(YY,XX,DR,DR(1,2),DR(1,3),DR(1,4),NP,IER)             ISS33400
      IF (IER .EQ. 1) GO TO 650                                        ISS33500
      XPL = (YY(NP)-YY(1))/200.0                                       ISS33600
      XPL2 = XPL*0.1                                                   ISS33700
      XP1 = XPL                                                        ISS33800
      J = 0                                                            ISS33900
      M = 1                                                            ISS34000
      YPL = YY(1)-XP1                                                  ISS34100
  631 YPL = YPL+XP1                                                    ISS34200
      IF (YPL .LE. X1-2.0*XPL) GO TO 632                               ISS34300
      XP1 = XPL2                                                       ISS34400
      IF (YPL .GE. X1+2.0*XPL) XP1 = XPL                               ISS34500
  632 IF (YPL .LT. YY(M+1)) GO TO 634                                  ISS34600
  633 M = M+1                                                          ISS34700
      IF (M .GE. NP) GO TO 670                                         ISS34800
      IF (YPL .GE. YY(M+1)) GO TO 633                                  ISS34900
  634 Y1 = YPL-YY(M)                                                   ISS35000
      Y1 = XX(M)+Y1*(DR(M,2)+(YPL-YY(M+1))*(2.*DR(M,3)+DR(M+1,3)+DR(M,1)ISS35100
     1*Y1)*.16666667)                                                  ISS35200
      J = J+1                                                          ISS35300
      DR(J,5) = Y1                                                     ISS35400
      DR(J,6) = YPL                                                    ISS35500
      IF (J .GE. 244) GO TO 670                                        ISS35600
      GO TO 631                                                        ISS35700
  650 DO 660 M=1,NP                                                    ISS35800
      DR(M,5) = XX(M)                                                  ISS35900
  660 DR(M,6) = YY(M)                                                  ISS36000
      NPI = NP                                                         ISS36100
  670 MP = 1                                                           ISS36200
      IF (IER .EQ. 0) NP = J                                           ISS36300
      IF (KB(L) .NE. 0) GO TO 680                                      ISS36400
      NP = NP+1                                                        ISS36500
      DR(NP,5) = DR(1,5)                                               ISS36600
      DR(NP,6) = DR(1,6)                                               ISS36700
  680 CONTINUE                                                         ISS36800
```

```
      DO 681 J=1,NP                                                     ISS36900
      I = J+MP-1                                                        ISS37000
      YPL = DR(I,6)*RAD                                                 ISS37100
      DR(I,6) = DR(I,5)*COS(YPL)+YSHFT                                  ISS37200
  681 DR(I,5) = DR(I,5)*SIN(YPL)+XSHFT                                  ISS37300
C     PLOT CURVE                                                        ISS37400
      IF (ISW3 .EQ. 2) GO TO 694                                        ISS37500
      M = NPI                                                           ISS37600
      CALL CALCS(XX,YY,M,RAD,RADI,X1,Y1,XSHFT,YSHFT,1)                  ISS37700
      IF (LINES .GT. 52) GO TO 686                                      ISS37800
      WRITE (6,2000) FI(N),(KLINE(J),J=1,NCV)                           ISS37900
      WRITE (6,2003)                                                    ISS38000
      LINES = LINES+3                                                   ISS38100
      GO TO 687                                                         ISS38200
  686 LINES = 57                                                        ISS38300
  687 CONTINUE                                                          ISS38400
      M1 = -5                                                           ISS38500
  688 M1 = M1+6                                                         ISS38600
      IF (M1 .GT. M) GO TO 692                                          ISS38700
      M2 = M1+5                                                         ISS38800
      IF (M2 .GT. M) M2 = M                                             ISS38900
      LINES = LINES+1                                                   ISS39000
      IF (LINES .LT. 57) GO TO 691                                      ISS39100
      IF (JM .GT. 1) GO TO 689                                          ISS39200
      CALL PRTTTL(NWD,LINES,LEGEND,0.0,0.0)                             ISS39300
      GO TO 690                                                         ISS39400
  689 CALL PRTTTL(NWD,LINES,LEGEND,DECAY,LAMBDA)                        ISS39500
  690 WRITE (6,2000) FI(N),(KLINE(J),J=1,NCV)                           ISS39600
      WRITE (6,2001)                                                    ISS39700
      LINES = LINES+7                                                   ISS39800
  691 WRITE (6,2002) (XX(J),YY(J),J=M1,M2)                              ISS39900
      GO TO 688                                                         ISS40000
  692 CONTINUE                                                          ISS40100
  694 CONTINUE                                                          ISS40200
      IF (ISW3 .EQ. 1) GO TO 695                                        ISS40300
      CALL ILPLOT(DR(MP,5),DR(MP,6),NP,1,DR)                            ISS40400
  695 CONTINUE                                                          ISS40500
      IP1 = IP1+NPP(L)                                                  ISS40600
  700 CONTINUE                                                          ISS40700
  710 CONTINUE                                                          ISS40800
      IF (ISW3 .EQ. 1) GO TO 800                                        ISS40900
```

```
      XPL = -XMIN*SCLX+XLM1-0.5*HT                                      ISS41000
      YPL = -YMIN*SCLY+YBM1-0.5*HT                                      ISS41100
      CALL PRINTV(1,1H*,IFIX(XPL),IFIX(YPL))                            ISS41200
  800 CONTINUE                                                          ISS41300
      RETURN                                                            ISS41400
 2000 FORMAT (1H0,40X,22H*-*-* ISOPLETH LEVEL =,F9.3,2H, ,9A6)          ISS41500
 2001 FORMAT(1H0,6(19H  RANGE    AZIMUTH )/1X,6(19H (METERS)  BEARING )/ISS41600
     11X,6(10X,9H(DEGREES))/1X,19(6H------))                            ISS41700
 2002 FORMAT (1X,6(F10.3,F8.3,1X))                                      ISS41800
 2003 FORMAT ()                                                         ISS41900
      END                                                               ISS42000
```

```
      SUBROUTINE ILPLOT(X,Y,N,ISW,IJ)                                   ILP00100
C     THIS SUBROUTINE PLOTS AND LABELS CURVES                           ILP00200
      COMMON /BNDS/ XRIT,XLFT,YBOT,YTOP,XPL,YPL                         ILP00300
      COMMON /ILPLTS/ XMAX,XMIN,YMAX,YMIN,XLM1,YBM1,HT,CHARF,SCLX,SCLY, ILP00400
     1XSIZE1,YSIZE1                                                     ILP00500
      DIMENSION X(1),Y(1),IJ(1)                                         ILP00600
      COMMON /XYXYPT/ YP(41),XP(41),A(41),B(41),C(41),D(41),XI(41),YI(41ILP00700
     1),NUM(3),NC                                                       ILP00800
      IFLG = 0                                                          ILP00900
      JFLG = 0                                                          ILP01000
      DO 100 I=1,N                                                      ILP01100
      X1 = X(I)                                                         ILP01200
      Y1 = Y(I)                                                         ILP01300
      IF (XLFT .LE. X1.AND.X1 .LE. XRIT) GO TO 20                       ILP01400
      GO TO 30                                                          ILP01500
   20 IF (YBOT .LE. Y1.AND.Y1 .LE. YTOP) GO TO 60                       ILP01600
   30 IF (JFLG .EQ. 0) GO TO 50                                         ILP01700
C     LAST POINT WAS OUT, THIS POINT IS OUT                             ILP01800
   40 IJ(I) = 3                                                         ILP01900
      GO TO 90                                                          ILP02000
   50 JFLG = 1                                                          ILP02100
      IF (I .EQ. 1) GO TO 40                                            ILP02200
C     THIS POINT OUT LAST POINT IN                                      ILP02300
C     INTERP FOR PLOT POINT                                             ILP02400
      CALL BOUNDS(X1,Y1,XLST,YLST)                                      ILP02500
      ITAG = 1                                                          ILP02600
      GO TO 80                                                          ILP02700
   60 IF (JFLG .EQ. 0) GO TO 70                                         ILP02800
C     THIS POINT IN LAST POINT OUT                                      ILP02900
      JFLG = 0                                                          ILP03000
      CALL BOUNDS(XLST,YLST,X1,Y1)                                      ILP03100
      IJ(I-1) = 0                                                       ILP03200
      X(I-1) = (XPL-XLFT)*SCLX+XLM1                                     ILP03300
      Y(I-1) = (YPL-YBOT)*SCLY+YBM1                                     ILP03400
   70 XPL = X1                                                          ILP03500
      YPL = Y1                                                          ILP03600
      ITAG = 2                                                          ILP03700
   80 X(I) = (XPL-XLFT)*SCLX+XLM1                                       ILP03800
      Y(I) = (YPL-YBOT)*SCLY+YBM1                                       ILP03900
      IJ(I) = ITAG                                                      ILP04000
```

```
   90 XLST = X1                                          ILP04100
      YLST = Y1                                          ILP04200
  100 CONTINUE                                           ILP04300
      IF (ISW .EQ. 2) GO TO 160                          ILP04400
C     IJ = 0 CURVE ENTERS GRAPH - FIRST POINT            ILP04500
C     IJ = 1 CURVE LEAVES GRAPH - LAST POINT             ILP04600
C     IJ = 2 CURVE CONTINUES WITHIN GRAPH                ILP04700
C     IJ = 3 CURVE OUTSIDE OF GRAPH DO NOT PLOT          ILP04800
C                                                        ILP04900
C     FIND POINTS FOR ISOPLETH LABELS                    ILP05000
C     FIND ALL POINTS WHERE CURVE LEAVES GRAPH           ILP05100
      M = 0                                              ILP05200
      DO 110 I=1,N                                       ILP05300
      IF (IJ(I) .NE. 1) GO TO 110                        ILP05400
      M = M+1                                            ILP05500
      B(M) = X(I)                                        ILP05600
      C(M) = Y(I)+.02                                    ILP05700
  110 CONTINUE                                           ILP05800
C     FIND ALL POINTS WHERE CURVE ENTERS GRAPH           ILP05900
      DO 120 I=1,N                                       ILP06000
      IF (IJ(I) .NE. 0) GO TO 120                        ILP06100
      M = M+1                                            ILP06200
      B(M) = X(I)                                        ILP06300
      C(M) = Y(I)+.02                                    ILP06400
  120 CONTINUE                                           ILP06500
      L = N/2                                            ILP06600
      IF (IJ(L) .EQ. 2) GO TO 130                        ILP06700
      L = 1                                              ILP06800
      IF (IJ(L) .EQ. 2) GO TO 130                        ILP06900
      L = N/4                                            ILP07000
      IF (IJ(L) .EQ. 2) GO TO 130                        ILP07100
      L = 3*N/4                                          ILP07200
      IF (IJ(L) .NE. 2) GO TO 140                        ILP07300
  130 M = M+1                                            ILP07400
      B(M) = X(L)                                        ILP07500
      C(M) = Y(L)+.02                                    ILP07600
  140 CONTINUE                                           ILP07700
C     PLOT LABELS                                        ILP07800
      XPL = -XLFT*SCLX+XLM1                              ILP07900
      YPL = -YBOT*SCLY+YBM1                              ILP08000
      DO 150 I=1,M                                       ILP08100
```

```
      IF (ABS(XPL-B(I)) .GT. 0.2) GO TO 145                                     ILP08200
      IF (ABS(YPL-C(I)) .LE. 0.2) GO TO 150                                     ILP08300
  145 CONTINUE                                                                  ILP08400
      CALL PRINTV(NC,NUM,IFIX(B(I)),IFIX(C(I)))                                 ILP08500
  150 CONTINUE                                                                  ILP08600
  160 CONTINUE                                                                  ILP08700
C     PLOT THE CURVE                                                            ILP08800
      IF (IJ(1) .NE. 3) IJ(1) = 0                                               ILP08900
      N = N-1                                                                   ILP09000
      DO 170 I=1,N                                                              ILP09100
      IF (IJ(I) .EQ. 3) GO TO 170                                               ILP09200
      IF (IJ(I+1) .EQ. 3) GO TO 170                                             ILP09300
      CALL LINE2V(IFIX(X(I)),IFIX(Y(I)),IFIX(X(I+1)-X(I)),IFIX(Y(I+1)-Y(ILP09400
     1I)))                                                                      ILP09500
  170 CONTINUE                                                                  ILP09600
  180 CONTINUE                                                                  ILP09700
      RETURN                                                                    ILP09800
      END                                                                       ILP09900
```

```
      SUBROUTINE CALCS(XX,YY,NP,RAD,RADI,XSHFT,YSHFT,X1,Y1,LLSW)        CAL00100
      DIMENSION XX(1),YY(1)                                             CAL00200
      IF (LLSW .EQ. 1) GO TO 5                                          CAL00300
      XSHFT = 0.0                                                       CAL00400
      YSHFT = 0.0                                                       CAL00500
    5 DO 10 I=1,NP                                                      CAL00600
      YPL = YY(I)*RAD                                                   CAL00700
      YY(I) = XX(I)*COS(YPL)+Y1                                         CAL00800
      XX(I) = XX(I)*SIN(YPL)+X1                                         CAL00900
      IF (LLSW .EQ. 1) GO TO 10                                         CAL01000
      XSHFT = XSHFT+XX(I)                                               CAL01100
      YSHFT = YSHFT+YY(I)                                               CAL01200
   10 CONTINUE                                                          CAL01300
      IF (LLSW .EQ. 1) GO TO 11                                         CAL01400
      XSHFT = XSHFT/FLOAT(NP)                                           CAL01500
      YSHFT = YSHFT/FLOAT(NP)                                           CAL01600
   11 CONTINUE                                                          CAL01700
      DO 20 I=1,NP                                                      CAL01800
      IF (LLSW .EQ. 1) GO TO 17                                         CAL01900
      XX(I) = XX(I)-XSHFT                                               CAL02000
      YY(I) = YY(I)-YSHFT                                               CAL02100
   17 CONTINUE                                                          CAL02200
      IF (XX(I)) 19,18,19                                               CAL02300
   18 IF (YY(I)) 19,20,19                                               CAL02400
   19 YPL = 90.0-ATAN2(YY(I),XX(I))*RADI                                CAL02500
      IF (YPL .LT. 0.0) YPL = YPL+360.0                                 CAL02600
      XX(I) = SQRT(XX(I)*XX(I)+YY(I)*YY(I))                             CAL02700
      YY(I) = YPL                                                       CAL02800
      IF (LLSW .EQ. 1) YY(I) = AMOD(YPL,360.0)                          CAL02900
   20 CONTINUE                                                          CAL03000
      IF (LLSW .EQ. 1) GO TO 31                                         CAL03100
      DO 30 M=2,NP                                                      CAL03200
      DO 30 I=2,NP                                                      CAL03300
      IF (YY(I) .GE. YY(I-1)) GO TO 30                                  CAL03400
      YPL = YY(I)                                                       CAL03500
      YY(I) = YY(I-1)                                                   CAL03600
      YY(I-1) = YPL                                                     CAL03700
      YPL = XX(I)                                                       CAL03800
      XX(I) = XX(I-1)                                                   CAL03900
      XX(I-1) = YPL                                                     CAL04000
```

```
   30 CONTINUE                                                              CAL04100
   31 RETURN                                                                CAL04200
      END                                                                   CAL04300
```

```
      SUBROUTINE MSFLD(I1,I2,IWRD,J1,JWRD)                              MSF00100
C     THIS PROG EXTRACTS AN I2 BIT BYTE FROM IWRD STARTING AT BIT I1 ANDMSF00200
C     STORES IT IN JWRD STARTING AT BIT J1. THE REMAINING BITS OF JWRD  MSF00300
C     ARE UNCHANGED. I1 AND J1 ARE COUNTED RIGHT FROM THE SIGN BIT AND  MSF00400
C     THE SIGN BIT IS BIT ZERO.                                         MSF00500
      FLD(J1,I2,JWRD) = FLD(I1,I2,IWRD)                                 MSF00600
      RETURN                                                            MSF00700
      END                                                               MSF00800
```

```
      FUNCTION RB11(PARM,P,Z,ZRK)                                       R1100100
      RB11 = PARM*(Z**P-ZRK**P)/(P*(Z-ZRK)*ZRK**(P-1.0))                R1100200
      RETURN                                                            R1100300
      END                                                               R1100400
```

```
      FUNCTION RB8(A,B,C)                                               RB800100
      RB8 = ALOG(A/B)*C                                                 RB800200
      IF (RB8+1.0) 20,10,20                                             RB800300
   10 RB8 = -.99999999                                                  RB800400
   20 RB8 = RB8+1.0                                                     RB800500
      RETURN                                                            RB800600
      END                                                               RB800700
```

```
      SUBROUTINE INPTS(KS,IB,NX,II,NY,D,T)                              INP00100
      DIMENSION D(41,1),T(1)                                            INP00200
      DO 10 I=1,NX                                                      INP00300
      KOUT = 4*I-4+KS-IB                                                INP00400
      CALL INTOUT(D,KOUT,NY,1,41,I)                                     INP00500
   10 CONTINUE                                                          INP00600
      IF (KS .LT. 5) GO TO 30                                           INP00700
      DO 20 I=1,NX                                                      INP00800
      KOUT = 4*I-2                                                      INP00900
      CALL INTOUT(T,KOUT,NY,1,1,1)                                      INP01000
      DO 20 J=1,NY                                                      INP01100
      TMP = 0.0                                                         INP01200
      IF (T(J) .LE. 0.0.OR.D(I,J) .LE. 0.0) GO TO 20                    INP01300
      TMP = T(J)/D(I,J)                                                 INP01400
   20 D(I,J) = TMP                                                      INP01500
   30 RETURN                                                            INP01600
      END                                                               INP01700
```

```
      SUBROUTINE INTOUT(D,I,N,L,IDM,K)                                  INT00100
      DIMENSION D(IDM,1)                                                INT00200
      INTEGER RECLTH                                                    INT00300
      IF (ISP .NE. 0) GO TO 10                                          INT00400
      ISP = 1                                                           INT00500
      RECLTH = 41                                                       INT00600
      DEFINE FILE 12(164,RECLTH,U,KOUK)                                 INT00700
   10 CONTINUE                                                          INT00800
      KOUK = I                                                          INT00900
   60 FORMAT (1X,8I6)                                                   INT01000
      GO TO (20,30),L                                                   INT01100
   20 READ (12'KOUK) (D(K,J),J=1,N)                                     INT01200
      GO TO 40                                                          INT01300
   30 WRITE (12'KOUK) (D(K,J),J=1,N)                                    INT01400
   40 RETURN                                                            INT01500
      END                                                               INT01600
```

APPENDIX D

PROGRAM OUTPUT FOR THE PREPROCESSOR PROGRAM AND THE NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM -- VERSION 5

D.1 PREPROCESSOR PROGRAM EXAMPLE OUTPUT

The Preprocessor Program produces both printed output and punched data-card output. The Program first prints a detailed list of all inputs including the constants pertaining to the rocket vehicle for which calculations are being made. The first page of the example output listing below shows the constants used for the example problem described in Section 5 of the main text for a simulated normal launch of a Space Shuttle vehicle at Kennedy Space Center on 21 October 1972. The Program next prints the FORTRAN NAMELIST data which are subsequently punched on cards for Model 4 calculations using the Multilayer Diffusion Models Program. In this example, the first printed list and punched card deck are for calculation of HCl concentrations and dosages; the second list and punched card deck are for CO concentrations and dosages; the third list and punched card deck are for Al_2O_3 concentrations and dosages. A printed list and punched card deck for the calculation of CO_2 concentrations and dosages are not produced for the Space Shuttle vehicle. The Program output for the Model 4 calculations ends with a summary table of the Program calculations. The Program then produces similar printed listings and sets of punched card decks in the same format for use in the Model 3 calculations.

--*-* NORMAL LAUNCHSPACE SHUTTLE VEHICLE

- INITIALIZED DATA USED FOR ABOVE VEHICLE *-*

QC - RATE OF OUTPUT OF EXHAUST MATERIAL FROM VEHICLE IN GRAMS/SEC IS 9.38498400+06
QT - TOTAL AMOUNT OF VEHICLE EXHAUST MATERIAL IN GRAMS IS 9.15626984+08
A AND B - VEHICLE RISE PARAMETERS IN THE EQUATION TR=A*Z**B ARE .663552 AND .485477
HEAT - TOTAL HEAT OUTPUT IN CALORIES/GRAM IS 2582.0000
GAMMA - ENTRAINMENT PARAMETER IS .6400
CP - SPECIFIC HEAT OF AIR IS .240
POLUTANT MATERIALS ARE HCL , CO , CO2 , AL203,
FRACTIONAL DISTRIBUTION OF THE ABOVE MATERIALS IS .207, .280, .000, .304,
MOLECULAR WEIGHT OF THE ABOVE MATERIALS IS 36.460, 28.010, 44.010,

*-X PROGRAM INPUT DATA *-*

DATA CARD 1
TITLE - KSC 21 OCT 72,
NORMAL - IS LAUNCH NORMAL ? YES
IFEET - ARE LAYER BOUNDARY HEIGHTS Z, AND HM IN FEET? NO
KNOTS - IS THE WIND SPEED WS IN KNOTS? NO
SIGAR - STANDARD DEVIATION OF THE AZIMUTH WIND ANGLE IN DEGREES IS 9.000
RHO - AIR DENSITY IN GRAMS/CUBIC METER IS 1197.070
HM - HEIGHT OF SURFACE MIXING LAYER IS 1432.000
DATA CARD 2 THROUGH 20

LAYER BOUNDARY Z (METERS)	WIND DIRECTION WD (DEG)	WIND SPEED WS (MET/S)	TEMPERATURE T (DEG C)	PRESSURE P (MB)	RH (PERCENT)
18.000	80.0000	6.0000	22.600	1022.000	57.000
194.000	81.0000	8.0000	22.200	1000.000	57.000
250.000	82.0000	9.0000	22.100	993.660	57.000
284.000	82.0000	10.0000	22.000	989.790	58.000
500.000	79.0000	10.0000	19.300	965.400	65.000
558.000	79.0000	10.0000	18.500	958.940	67.000
637.000	78.0000	10.0000	17.800	950.000	70.000
750.000	76.0000	11.0000	16.900	937.710	74.000
1000.000	71.0000	11.0000	14.600	910.610	86.000
1098.000	68.0000	11.0000	13.700	900.000	93.000
1135.000	67.0000	11.0000	13.300	896.250	94.000
1250.000	63.0000	11.0000	12.300	884.090	97.000
1432.000	56.0000	11.0000	10.700	865.130	97.000
1500.000	53.0000	10.0000	10.500	858.170	90.000
1577.000	49.0000	10.0000	10.300	850.000	79.000
1716.000	40.0000	9.0000	9.900	836.310	55.000
1750.000	37.0000	8.0000	10.300	832.310	55.000
2000.000	9.0000	6.0000	12.500	808.340	49.000
2259.000	344.0000	5.0000	11.100	783.690	44.000
2500.000	342.0000	5.0000	9.100	761.390	54.000

--*-* EXAMPLE SPACE SHUTTLE NORMAL LAUNCH *-*-*-*
--*-* NORMAL LAUNCH SPACE SHUTTLE VEHICLE *-*-*-*

- NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

```
$NAM2
TESTNO=60HKSC 21 OCT 72,                          SPACE SHUTTLE
NAMCAS=60H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,1,2,0,0,0,0,NPS= 0,NZS=13,NDI=69,NCI=69,NTI=62,ZRK= 16.0,
TAUK= 447.273,IZMOD=4,4,4,4,4,4,4,4,4,4,4,4,
    Z=      .000,  194.000,  250.000,  284.000,  500.000,  556.000,  637.000,
  750.000, 1000.000, 1098.000, 1135.000, 1250.000, 1432.000,
    U=  4.4227630+07,  1.7676125+07,  1.3832882+07,  1.7371957+06,
 6.5352765+07,  1.5651766+08,  3.3045618+08,  1.4067362+09,  9.0119715+08,
  4.0267076+08,  1.4819777+09,  3.0778224+09,
 UBARK=    6.000,    8.000,    9.000,   10.000,   10.000,   10.000,   10.000,
   11.000,   11.000,   11.000,   11.000,   11.000,   11.000,
 SIGAK=    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
 SIGEK=    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
 SIGXO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGYO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGZO=     .000,     .000,     .000,     .000,     .000,     .000,     .000,
     .000,     .000,     .000,     .000,     .000,
THETAK=   80.000,   81.000,   82.000,   82.000,   79.000,   79.000,   78.000,
   76.000,   71.000,   66.000,   67.000,   63.000,   56.000,
TIMAV= 600.0,
    DI=  400.000,  200.000,  100.000,   50.000,   25.000,    5.000,
    CI=   16.000,    8.000,    4.000,    1.000,     .500,     .100,
    TI=   30.000,    4.000,    8.000,    2.000,    1.000,     .500,
  DELX=   39.922,   67.179,   88.489,  279.077,  343.023,  438.965,  601.038,
 1055.551, 1259.946, 1340.757, 1610.334, 2083.997,
  DELY=  260.300,  260.906,  261.169,  260.712,  260.393,  259.979,  259.175,
  256.734,  255.544,  255.057,  253.360,  250.167,
 TEMPK=  295.624,  297.076,  297.521,  297.781,  297.137,  296.876,  296.979,
  297.192,  297.393,  297.523,  297.445,  297.538,  297.580,
H= 1790.000,
$END
```

```
*-*-*-*  EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                                    *-*-*-*
      *-*-*-*      NORMAL LAUNCH SPACE SHUTTLE     VEHICLE      *-*-*-*

   *-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

$NAM2
TESTNO=60HKSC 21 OCT 72,                         SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,2,2,0,0,0,0,NPS= 0,NZS=13,NDI=69,NCI=69,NTI=61,ZRK= 18.0,
TAUK= 447.273,IZMOD=4,4,4,4,4,4,4,4,4,4,4,4,4,
     Z=      .000,  194.000,  250.000,  284.000,  500.000,  558.000,  637.000,
  750.000, 1000.000, 1098.000, 1135.000, 1250.000, 1432.000,
     Q=  7.7872643+07,  3.1474920+07,  2.4355885+07,  3.0587220+08,
  1.5028265+08,  2.7558477+08,  5.8184208+08,  2.4768740+09,  1.5867593+09,
  7.0793558+08,  2.6093535+09,  5.4191954+09,
 UBARK=     6.000,     8.000,     9.000,    10.000,    10.000,    10.000,    10.000,
   11.000,    11.000,    11.000,    11.000,    11.000,    11.000,
 SIGAK=     4.500,     4.500,     4.500,     4.500,     4.500,     4.500,     4.500,
    4.500,     4.500,     4.500,     4.500,     4.500,     4.500,
 SIGEK=     4.500,     4.500,     4.500,     4.500,     4.500,     4.500,     4.500,
    4.500,     4.500,     4.500,     4.500,     4.500,     4.500,
 SIGXO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGYO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGZO=      .000,      .000,      .000,      .000,      .000,      .000,      .000,
     .000,      .000,      .000,      .000,      .000,
THETAK=    80.000,    81.000,    82.000,    82.000,    79.000,    79.000,    78.000,
   76.000,    71.000,    68.000,    67.000,    63.000,    56.000,
TIMAV= 360.0,
    DI=  400.000,  200.000,  100.000,   50.000,   25.000,    5.000,
    CI=   35.000,   10.000,    4.000,    2.000,    1.000,     .100,
    TI=  150.000,  100.000,   60.000,   30.000,   15.000,    1.000,
  DELX=    39.922,    67.179,    88.489,  279.077,  343.023,  438.965,  601.038,
 1053.551, 1259.946, 1340.737, 1610.334, 2083.997,
  DELY=  260.500,  260.906,  261.169,  260.712,  260.393,  259.979,  259.175,
  256.734,  255.544,  255.057,  253.360,  250.167,
 TEMPK=  295.624,  297.076,  297.521,  297.781,  297.137,  296.876,  296.979,
  297.192,  297.393,  297.523,  297.445,  297.538,  297.580,
H= 1790.000,
$END
```

```
            *-*-*-*  EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                       *-*-*-*
                 *-*-*-*      NORMAL LAUNCH SPACE SHUTTLE     VEHICLE     *-*-*-*

                  *-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

$NAM2
TESTNO=60HKSC 21 OCT 72,                           SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,4,2,0,0,0,0,NPS= 0,NZS=13,NDI=69,NCI=69,NTI=61,ZRK= 18.0,
TAJK= 447,273,IZMOD=4,4,4,4,4,4,4,4,4,4,4,4,
     Z=      .000,  194.000,  250.000,  284.000,  500.000,  558.000,  637.000,
  750.000, 1000.000, 1098.000, 1135.000, 1250.000, 1432.000,
     Q=  9.8491545+07,  3.9808761+07,  3.0804768+07,  3.8686018+08,
  1.9007407+08,  3.4855334+08,  7.5590061+08,  3.1326938+09,  2.0068970+09,
  8.9538080+08,  3.3002509+09,  6.8540749+09,
 UBARK=    6.000,    8.000,    9.000,   10.000,   10.000,   10.000,   10.000,
   11.000,   11.000,   11.000,   11.000,   11.000,   11.000,
 SIGAK=    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
 SIGEK=    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
    4.500,    4.500,    4.500,    4.500,    4.500,    4.500,
 SIGXO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGYO=  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,  532.837,
  532.837,  532.837,  532.837,  532.837,  532.837,
 SIGZO=     .000,     .000,     .000,     .000,     .000,     .000,     .000,
     .000,     .000,     .000,     .000,     .000,
THETAK=   80.000,   81.000,   82.000,   82.000,   79.000,   79.000,   78.000,
   76.000,   71.000,   68.000,   67.000,   63.000,   56.000,
TIMAV= 600.0,
    DI=   40.000,   20.000,   10.000,    5.000,    2.500,     .500,
    CI=    2.000,    1.000,     .400,     .100,     .050,     .010,
    TI=   50.000,  100.000,   25.000,   10.000,    5.000,    1.000,
  DELX=   39.922,   67.179,   88.489,  279.077,  343.023,  438.965,  601.038,
 1053.551, 1259.946, 1340.737, 1610.334, 2083.997,
  DELY=  260.500,  260.906,  261.169,  260.712,  260.393,  259.979,  259.175,
  256.734,  255.544,  255.057,  253.360,  250.167,
 TEMPK=  295.624,  297.076,  297.521,  297.781,  297.137,  296.876,  296.979,
  297.192,  297.393,  297.523,  297.445,  297.538,  297.580,
H= 1790.000,
$END
```

--*-* EXAMPLE SPACE SHUTTLE NORMAL LAUNCH *-*-*-*
--*-* NORMAL LAUNCH SPACE SHUTTLE VEHICLE *-*-*-*

- CALCULATED PARAMETERS FOR MODEL 4 *-*

H - ADJUSTED CLOUD HEIGHT IN METERS IS 1790.000
ZM - REAL CLOUD HEIGHT IN METERS IS 1790.000 .
TAUK - TIME TO CLOUD STABILIZATION IN SEC IS 447.273
DPDZ - VERTICAL GRADIENT OF AMBIENT POTENTIAL TEMP IN DEGREES K/METER IS .001489
JBOT - BOTTOM LAYER FOR USE WITH MODEL 4 IS 1
JTOP - TOP LAYER FOR USE WITH MODEL 4 IS 12
Z - BOUNDARY HEIGHT AT THE BOTTOM OF LAYER 1 IN METERS IS .000
SIGAP - STANDARD DEVIATION OF THE WIND AZIMUTH ANGLE AT THE MEASUREMENT HEIGHT ZRK= 18.00 METERS IS 4.500
SIGEP - STANDARD DEVIATION OF THE WIND ELEVATION ANGLE AT ZRK IS 4.500

LAYER PARAMETERS -

LAYER NO.	Z (LAYER TOP)	HCL (SOURCE STRENGTH Q)	CO	CO2	AL2O3	SIGAP (DEG)	SIGEP (DEG)	SIGXO (METER)	SIGYO (METER)	SIGZO (METER)	DELX (METER)	DELY (DEG)
1	194.000	4.4227630+07	7.7872643+07	0.0000000	9.8491545+07	4.5000	4.5000	532.8372	532.8372	.0000	39.92	260.50
2	250.000	1.7876125+07	3.1474920+07	0.0000000	3.9808761+07	4.5000	4.5000	532.8372	532.8372	.0000	67.18	260.91
3	284.000	1.3832882+07	2.4355885+07	0.0000000	3.0804768+07	4.5000	4.5000	532.8372	532.8372	.0000	88.49	261.17
4	500.000	1.7371957+08	3.0587220+08	0.0000000	3.8686018+08	4.5000	4.5000	532.8372	532.8372	.0000	279.08	260.71
5	558.000	8.5352765+07	1.5028265+08	0.0000000	1.9007407+08	4.5000	4.5000	532.8372	532.8372	.0000	343.02	260.39
6	637.000	1.5651768+08	2.7558477+08	0.0000000	3.4855334+08	4.5000	4.5000	532.8372	532.8372	.0000	438.96	259.98
7	750.000	3.3045618+08	5.8184208+08	0.0000000	7.3590061+08	4.5000	4.5000	532.8372	532.8372	.0000	601.04	259.18
8	1000.000	1.4067362+09	2.4768740+09	0.0000000	3.1326938+09	4.5000	4.5000	532.8372	532.8372	.0000	1053.55	256.73
9	1098.000	9.0119715+08	1.5867593+09	0.0000000	2.0068970+09	4.5000	4.5000	532.8372	532.8372	.0000	1259.95	255.54
10	1135.000	4.0207076+08	7.0793558+08	0.0000000	8.9538080+08	4.5000	4.5000	532.8372	532.8372	.0000	1340.74	255.06
11	1250.000	1.4819777+09	2.6093535+09	0.0000000	3.3002509+09	4.5000	4.5000	532.8372	532.8372	.0000	1610.33	253.36
12	1432.000	3.0778224+09	5.4191954+09	0.0000000	6.8540749+09	4.5000	4.5000	532.8372	532.8372	.0000	2084.00	250.17
13	1500.000	1.3646208+09	2.4027204+09	0.0000000	3.0389060+09	.1000	.1000	532.8372	532.8372	.0000	2278.11	248.80
14	1577.000	1.6626373+09	2.9274452+09	0.0000000	3.7025659+09	.1000	.1000	532.8372	532.8372	.0000	2516.03	247.07
15	1716.000	3.2292515+09	5.6858202+09	0.0000000	7.1912958+09	.1000	.1000	532.8372	532.8372	.0000	3001.70	243.27
16	1750.000	8.1639239+08	1.4374416+09	0.0000000	1.8180433+09	.1000	.1000	532.8372	532.8372	.0000	3246.75	241.29
17	2000.000	7.6941059+09	1.3547196+10	0.0000000	1.7134185+10	.1000	.1000	507.5349	507.5349	.0000	4103.23	235.46
18	2259.000	7.1501190+09	1.2569385+10	0.0000000	1.5922767+10	.1000	.1000	431.7767	431.7767	.0000	2460.00	176.50
19	2500.000	4.9842630+09	8.7759108+09	0.0000000	1.1099572+10	.1000	.1000	357.3581	357.3581	.0000	2236.36	163.00

```
            *-*-*-*  EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                              *-*-*-*
                *-*-*-*      NORMAL LAUNCH SPACE SHUTTLE     VEHICLE       *-*-*-*

                   *-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

$NAM2
TESTNO=60HKSC 21 OCT 72,                          SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,1,2,0,0,0,0,NPS= 0,NZS= 2,NDI=69,NCI=69,NTI=62,ZRK= 18.0,
TAUK= 447.273,IZMOD=3,
     Z=     .000, 1432.000,
     Q=  8.0917873+09,
 UBARK=    6.000,   11.000,
 SIGAK=    4.500,    4.500,
 SIGEK=    4.500,    4.500,
 SIGXO=  532.837,
 SIGYO=  532.837,
 SIGZO=  183.163,
THETAK=   80.000,   56.000,
TIMAV= 600.0,
    DI=  400.000,  200.000,  100.000,   50.000,   25.000,    5.000,
    CI=   16.000,    8.000,    4.000,    1.000,     .500,     .100,
    TI=   30.000,    4.000,    8.000,    2.000,    1.000,     .500,
  DELX= 4103.229,
  DELY=  235.455,
 TEMPK=  295.624,  297.076,
H= 1038.200,
$END
```

```
             *-*-*-*  EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                                   *-*-*-*
                 *-*-*-*      NORMAL LAUNCH SPACE SHUTTLE     VEHICLE      *-*-*-*

                   *-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

$NAM2
TESTNO=60HKSC 21 OCT 72,                               SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,2,2,2,0,0,0,0,NPS= 0,NZS= 2,NDI=69,NCI=69,NTI=61,ZRK= 18.0,
TAUK= 447.273,IZMOD=3,
     Z=     .000, 1432.000,
     Q=  1.4247403+10,
 UBARK=    6.000,   11.000,
 SIGAK=    4.500,    4.500,
 SIGEK=    4.500,    4.500,
 SIGXO=  532.837,
 SIGYO=  532.837,
 SIGZO=  183.163,
THETAK=   80.000,   56.000,
TIMAV= 360.0,
    DI=  400.000,  200.000,  100.000,   50.000,   25.000,    5.000,
    CI=   35.000,   10.000,    4.000,    2.000,    1.000,     .100,
    TI=  150.000,  100.000,   60.000,   30.000,   15.000,    1.000,
  DELX= 4103.229,
  DELY=  235.455,
 TEMPK=  295.624,  297.076,
H= 1038.200,
$END
```

```
*-*-*-*  EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                                    *-*-*-*
    *-*-*-*      NORMAL LAUNCH SPACE SHUTTLE     VEHICLE      *-*-*-*

      *-* NAMELIST NAM2 FOR INPUT TO THE NASA/MSFC MULTILAYER MODEL VERSION 5 *-*

$NAM2
TESTNO=60HKSC 21 OCT 72,                         SPACE SHUTTLE
NAMCAS=68H EXAMPLE SPACE SHUTTLE NORMAL LAUNCH
ISKIP=0,3,3,0,4,2,0,0,0,0,NPS= 0,NZS= 2,NDI=69,NCI=69,NTI=61,ZRK= 18.0,
TAUK= 447.273,IZMOD=3,
     Z=     .000, 1432.000,
     Q=  1.8019791+10,
 UBARK=    6.000,   11.000,
 SIGAK=    4.500,    4.500,
 SIGEK=    4.500,    4.500,
 SIGXO=  532.837,
 SIGYO=  532.837,
 SIGZO=  183.163,
THETAK=   80.000,   56.000,
TIMAV= 600.0,
    DI=   40.000,   20.000,   10.000,    5.000,    2.500,     .500,
    CI=    2.000,    1.000,     .400,     .100,     .050,     .010,
    TI=   50.000,  100.000,   25.000,   10.000,    5.000,    1.000,
  DELX= 4103.229,
  DELY=  235.455,
 TEMPK=  295.624,  297.076,
H= 1038.200,
$END
```

--*-* EXAMPLE SPACE SHUTTLE NORMAL LAUNCH *-*-*-*
--*-* NORMAL LAUNCH SPACE SHUTTLE VEHICLE *-*-*-*

- CALCULATED PARAMETERS FOR MODEL 3 *-*

H - ADJUSTED CLOUD HEIGHT IN METERS IS 1038.200
ZM - REAL CLOUD HEIGHT IN METERS IS 1790.000
TAUK - TIME TO CLOUD STABILIZATION IN SEC IS 447.273
DPDZ - VERTICAL GRADIENT OF AMBIENT POTENTIAL TEMP IN DEGREES K/METER IS .001489
JBOT - BOTTOM LAYER FOR USE WITH MODEL 4 IS 1
JTOP - TOP LAYER FOR USE WITH MODEL 4 IS 12
Z - BOUNDARY HEIGHT AT THE BOTTOM OF LAYER 1 IN METERS IS .000
SIGAP - STANDARD DEVIATION OF THE WIND AZIMUTH ANGLE AT THE MEASUREMENT HEIGHT ZRK= 18.00 METERS IS 4.500
SIGEP - STANDARD DEVIATION OF THE WIND ELEVATION ANGLE AT ZRK IS 4.500

LAYER PARAMETERS -

LAYER NO.	Z (LAYER TOP)	- SOURCE STRENGTH Q - HCL	CO	CO2	AL2O3	SIGAP (DEG)	SIGEP (DEG)	SIGXO (METER)	SIGYO (METER)	SIGZO (METER)	DELX (METER)	DELY (DEG)
1	1432.000	8.0917873+09	1.4247403+10	0.0000000	1.8019791+10	4.5000	4.5000	532.8372	532.8372	183.1628	4103.23	235.46

D.2 NASA/MSFC MULTILAYER DIFFUSION MODELS PROGRAM EXAMPLE OUTPUT

This section contains a listing of the output produced by the NASA/MSFC Multilayer Diffusion Models Program using the example input data shown in Figure B-1, Appendix B and that portion of the data deck produced by the Preprocessor Program for HCl concentrations for a normal launch of a Space Shuttle vehicle. The input data shown in Figure B-1 instructed the Multilayer Diffusion Models Program to use Models 3 and 4 to calculate maximum centerline HCl concentrations, dosages and time-mean concentrations and to calculate the corresponding isopleths.

As shown below, the Program first lists a title page containing the data set title, the meteorological case, date and time, and the height, range and azimuth bearing of the stabilized cloud. Next, the Program prints the data parameters for each layer and the results of the parameter calculations (for example, mean wind speed, wind speed shears, etc.) in each layer. The inputs for Layer 1 are shown as the second page of the example listing. This page is followed by a list of maximum centerline concentrations, dosages, time-mean concentrations, the time of cloud passage and average concentration at various distances (ranges) along the cloud trajectory (azimuth bearing). The Program then produces a series of pages containing printer plots on log-log scales of the maximum centerline concentrations, dosages and time-mean concentrations at selected distances along the cloud trajectory.

After the printer plots are generated, the Program lists the range and azimuth bearing from the launch pad to selected isopleths of concentration, dosage and time-mean concentration. The printed ranges and azimuth bearings are ordered such that the output values begin at a range close to the launch pad and move in an azimuth direction clockwise around the isopleth. The listing for the Model 4 calculations ends with a printout of model input parameters for the layers that contributed to the ground-level concentrations and dosages previously listed. The computer Program then lists similar output for the Model 3 calculations except that model input parameters are given only for the sample layer used in the calculations.

The NASA/MSFC Multilayer Diffusion Models Program also contains provision for generating plots of maximum centerline concentrations, dosages, time-mean concentrations and isopleths of these parameters. Figures D-1 and D-2 respectively show computer plots of the Model 3 and Model 4 calculations of maximum centerline concentrations for the example case. Figures D-3 and D-4 respectively show plots of Model 3 and Model 4 HCl dosage isopleths for the example case.

```
*****************************************************************************************
*                                                                                       *
*     EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                                               *
*                                                                                       *
*                      KSC 21 OCT 72.                                                   *
*                                                                                       *
*                         DATE = 07/17/75, TIME = 16/07/48                              *
*                                                                                       *
*                                                                                       *
* ADJUSTED CLOUD RISE HEIGHT = 1790.00, RANGE =  2084.00, AZIMUTH BEARING = 250.17      *
*                                                                                       *
*****************************************************************************************
```

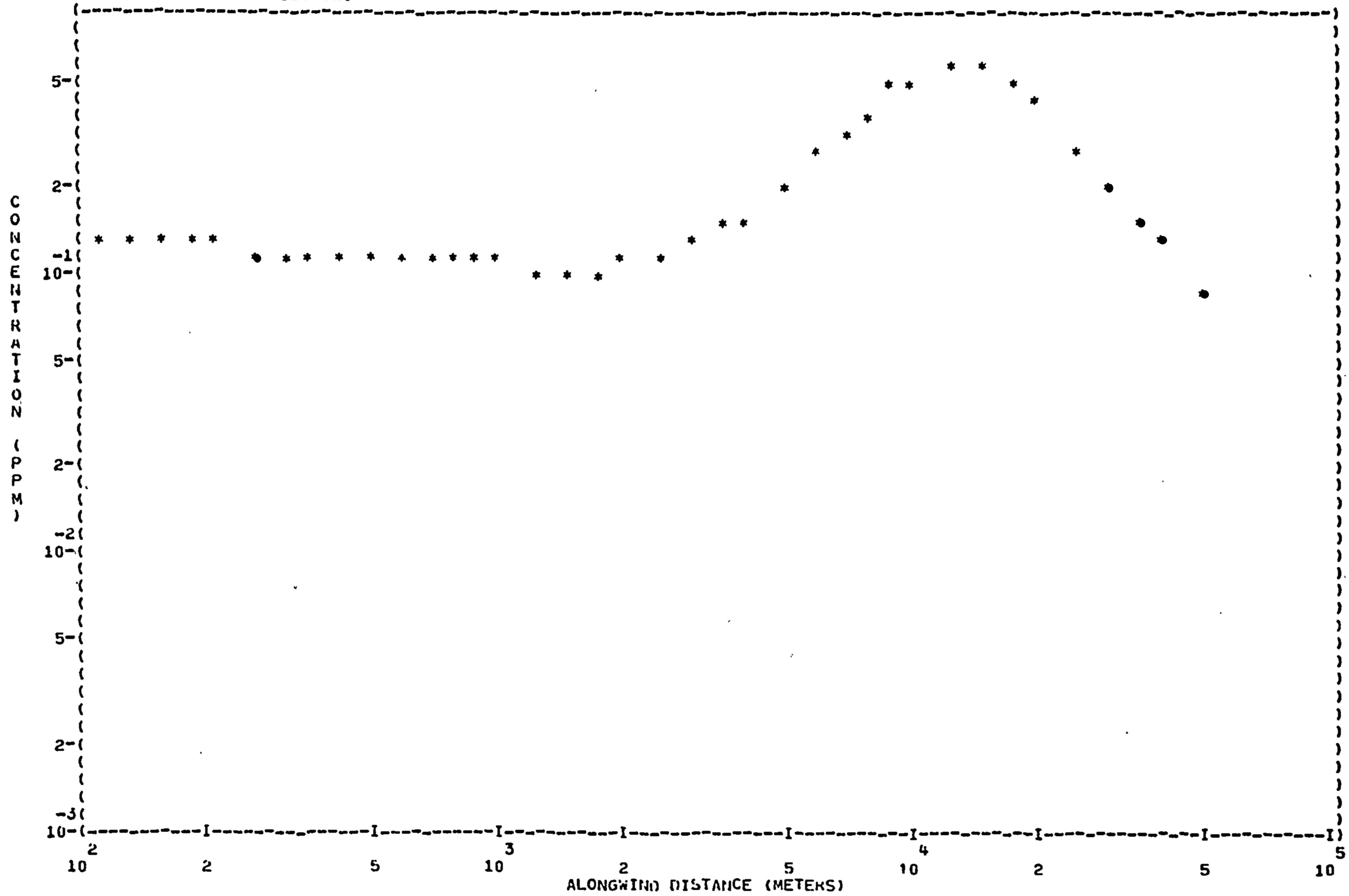
```
                              ***** LAYER 1 *****

                               ** INPUT DATA **

 Q=  .44227627+08, ZRK= 18.000, UBAR AT BOTTOM=  6.0000, UBAR AT TOP=  8.0000, SIGAK AT BOTTOM= 4.50000
SIGAK AT TOP= 4.50000, SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, TAUK= 447.273, TAUOK= 447.273
SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  80.000, THETAK AT TOP=  81.000, Z=     .000
ALPHA= 1.0 BETA= 1.0, H= 1790.000, DELX=  .39922000+02, DELY=  .26050000+03, IZMOD=  4, TIM1=  .00000000
ZLIM=     .000, LAMBDA=  .0000, TIMAV= 600.000, XRY= 100.000, XPZ= 100.000, XLRY=    .000, XLRZ=    .000, GAMMAP= .000

 ZRL= 18.000, UBARL AT BOTTOM=  6.0000, UBARL AT TOP= 11.0000, SIGAL AT BOTTOM= 4.50000, SIGAL AT TOP= 4.50000
SIGEL AT BOTTOM= 4.50000, SIGEL AT TOP= 4.50000, THETAL AT BOTTOM=  80.000, THETAL AT TOP=  56.000, TAUL= 447.273
TAUOL= 447.273, ALPHL= 1.0, BETL= 1.0, TAST=  .00000000   , JHOT= 1, JTOP=12

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =   7.31893, THETA =  80.50000, DELTHP =   1.00000, DELU =   2.00000
, SIGAP =    .07854, SIGEP =    .07854

CALCULATED INPUT PARAMETERS FOR LAYER CHANGE MODEL 4 *** UBAR =  10.03876, THETA =  68.00000, DELTHP = -24.00000
  DELU =   5.00000, SIGAP =    .07854, SIGEP =    .07854
          ---------------------------------------------------------------------------------------------------------
```

MAXIMUM CENTERLINE HCL CALCULATIONS AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

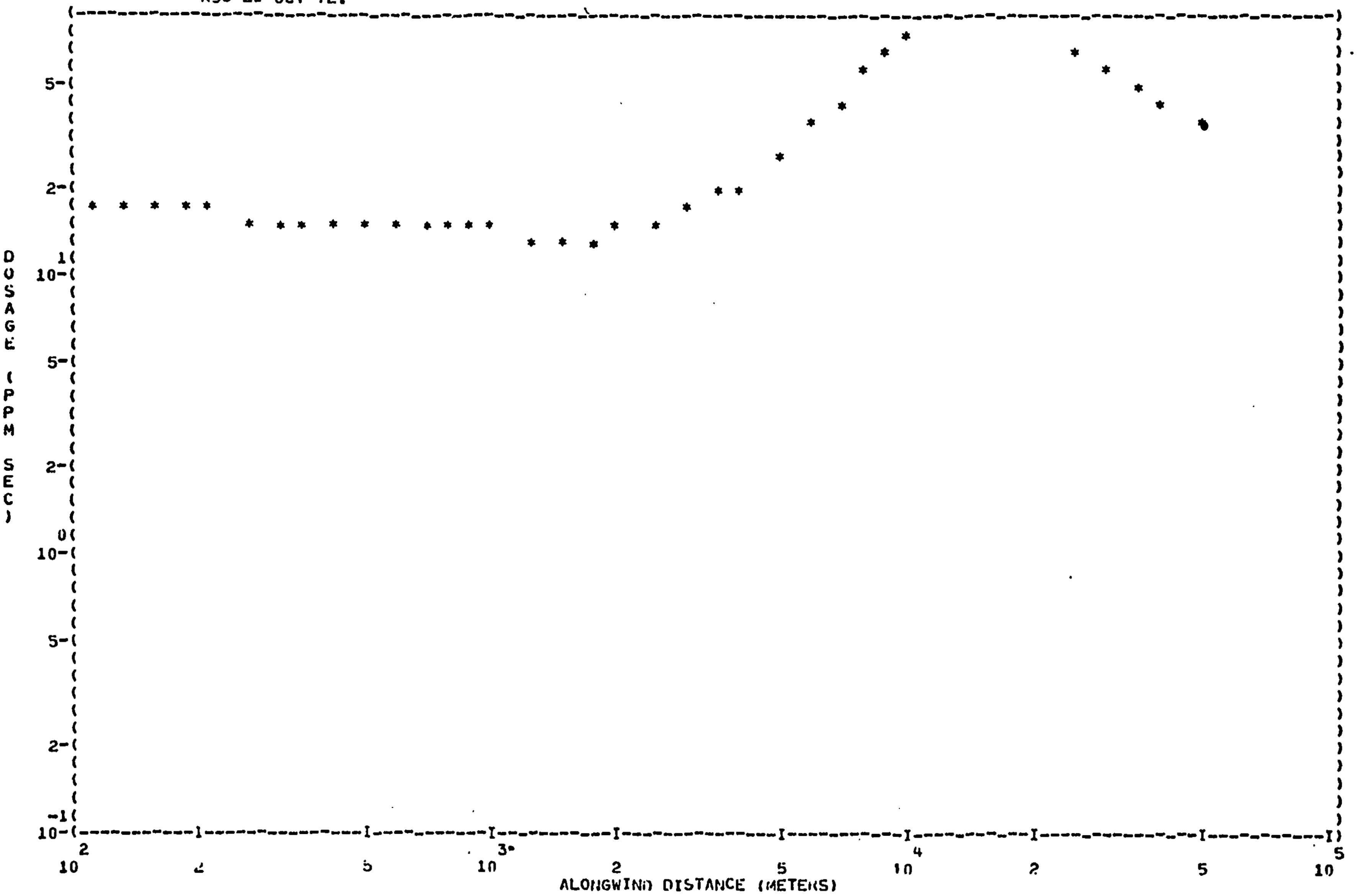
.597 IS THE MAXIMUM PEAK CONCENTRATION
97.569 IS THE MAXIMUM DOSAGE
.163 IS THE MAXIMUM PEAK 10.0 MINUTE TIME-MEAN CONCENTRATION

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	MAXIMUM PEAK CONCENTRATION (PPM)	MAXIMUM DOSAGE (PPM SEC)	MAXIMUM PEAK 10.0 MINUTE TIME-MEAN CONCENTRATION (PPM)	TIME OF CLOUD PASSAGE (SECONDS)	AVERAGE CLOUD CONCENTRATION (PPM)
50.0	284.2	.128	16.322	.027	228.235	.072
100.0	256.6	.127	16.852	.028	228.235	.074
125.0	254.1	.126	16.789	.028	228.236	.074
150.0	252.7	.126	16.727	.028	228.237	.073
175.0	251.8	.125	16.665	.028	228.239	.073
200.0	251.2	.125	16.603	.028	228.241	.073
250.0	250.4	.124	16.479	.027	228.247	.072
300.0	250.0	.123	16.356	.027	228.236	.072
350.0	249.7	.122	16.234	.027	228.235	.071
400.0	249.4	.121	16.113	.027	228.237	.071
500.0	249.1	.119	15.872	.026	228.237	.070
600.0	248.9	.117	15.634	.026	228.235	.069
700.0	248.8	.116	15.400	.026	228.240	.067
800.0	248.7	.114	15.168	.025	228.254	.066
900.0	248.6	.112	14.945	.025	228.276	.065
1000.0	248.5	.111	14.738	.025	228.307	.065
1250.0	248.4	.107	14.309	.024	228.232	.063
1500.0	248.4	.105	14.047	.023	228.247	.062
1750.0	248.4	.105	14.069	.023	228.244	.062
2000.0	248.5	.106	14.420	.024	228.302	.063
2500.0	248.6	.117	15.785	.026	228.309	.069
3000.0	248.7	.130	17.553	.029	228.591	.077
3500.0	248.7	.144	19.559	.033	229.083	.085
4000.0	248.8	.160	21.816	.036	229.784	.095
5000.0	248.9	.201	27.765	.046	231.605	.120
6000.0	249.0	.256	35.829	.060	234.631	.153
7000.0	249.0	.318	45.247	.075	238.235	.190
8000.0	248.9	.387	55.976	.093	242.584	.231
9000.0	248.8	.458	67.608	.113	247.637	.273
10000.0	248.7	.522	78.636	.131	253.351	.310
12500.0	248.6	.597	95.777	.160	270.245	.354
15000.0	248.5	.567	97.569	.163	290.329	.336
17500.0	248.4	.492	91.376	.152	312.989	.292
20000.0	248.3	.415	82.976	.136	337.707	.246
25000.0	248.3	.293	68.043	.113	391.725	.174
30000.0	248.2	.215	57.279	.095	450.003	.127
35000.0	248.2	.164	49.428	.081	511.082	.097
40000.0	248.2	.128	43.461	.071	574.070	.076
50000.0	248.1	.084	35.005	.054	703.713	.050

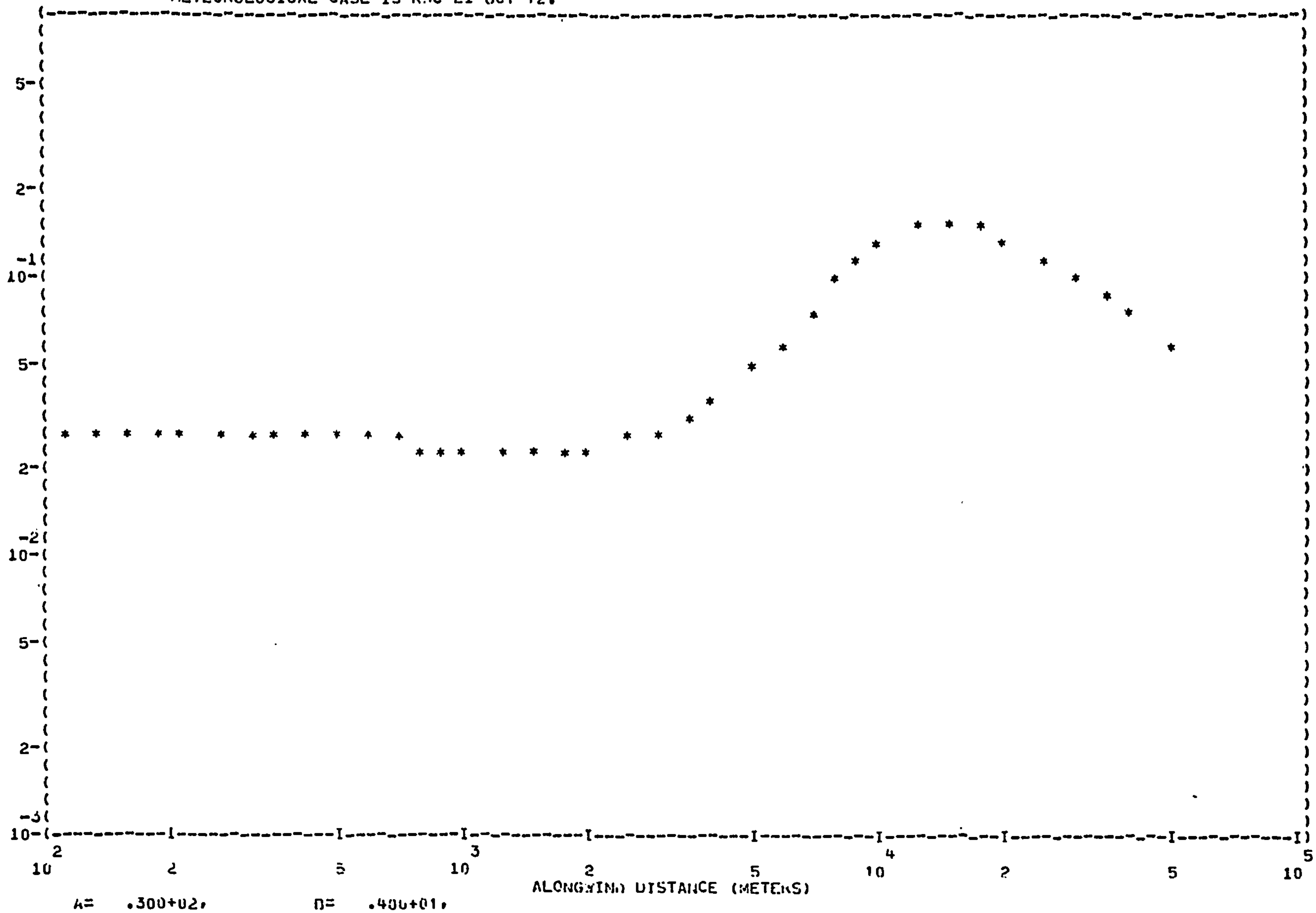
MAXIMUM CENTERLINE HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



MAXIMUM CENTERLINE HCL DOSAGE IN PPM SEC AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



MAXIMUM CENTERLINE HCL 10 MINUTE TIME-MEAN CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



A= .300+02, B= .400+01,

ISOPLETHS HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

--* ISOPLETH LEVEL = .500, CONCENTRATION (PPM)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
9740.952	247.500	10000.000	246.548	10972.021	245.000	12500.000	243.857	15000.000	244.577	15560.100	245.000
17131.179	247.500	10898.565	250.000	15000.000	252.396	14810.763	252.500	12500.000	253.276	11233.255	252.500
10000.000	250.869	9749.320	250.000								

--* ISOPLETH LEVEL = .100, CONCENTRATION (PPM)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
50.001	209.461	923.045	230.000	3000.001	237.762	3500.001	237.007	4000.001	236.489	5000.001	235.655
5977.487	235.000	7000.000	234.450	8000.001	234.037	9000.000	233.702	10000.000	233.473	12500.000	233.384
15000.000	233.803	17500.000	234.559	18767.787	235.000	25000.000	237.121	30000.000	238.864	35000.000	240.759
39299.485	242.500	44325.712	245.000	46367.489	247.500	45756.911	250.000	42311.098	252.500	36809.623	255.000
30196.775	257.500	25000.000	259.443	20000.000	261.358	17500.000	262.210	15000.000	263.076	12500.000	263.732
10000.000	263.921	9000.000	263.836	8000.000	263.654	7000.000	263.355	6000.000	262.829	5556.863	262.500
5000.000	262.012	4000.000	260.992	3500.000	260.386	3000.000	259.514	2500.001	258.142	1102.956	262.500

ISOPLETHS HCL DOSAGE IN PPM SEC AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

--* ISOPLETH LEVEL = 50.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
7891.711	245.000	6776.813	242.500	10000.000	240.757	11251.232	240.000	12500.000	239.391	15000.001	239.356
17500.000	239.873	17936.565	240.000	20000.000	240.621	25000.000	242.325	30000.000	244.401	34498.688	247.500
33719.494	250.000	30000.000	252.064	25000.000	254.182	23184.092	255.000	20000.000	256.082	17500.000	256.911
15148.421	257.500	15000.000	257.538	12500.000	257.700	11950.255	257.500	10000.000	256.638	9000.000	255.451
8300.000	253.050	7475.542	250.000								

--* ISOPLETH LEVEL = 25.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
5000.001	243.889	7000.001	238.583	9000.001	236.600	12500.001	235.436	15000.000	235.522	17500.000	235.906
20000.000	238.434	24561.625	237.500	25000.000	237.599	30000.000	238.552	35000.000	239.483	37974.351	240.000
40000.000	240.380	50000.000	242.096	64543.615	245.000	71029.357	247.500	68744.104	250.000	59812.486	252.500
50000.000	254.147	40035.337	255.000	40000.000	255.983	34999.999	256.900	32000.411	257.500	29999.999	257.895
24999.999	259.009	20763.984	260.000	20000.000	260.192	17500.000	260.881	15000.000	261.428	12500.000	261.696
10000.000	261.375	6000.001	260.249	6000.000	257.553	4753.132	252.500				

--* ISOPLETH LEVEL = 5.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
1191.870	200.000	3000.001	224.765	4000.001	227.024	5000.001	228.148	6000.001	228.671	7000.001	228.954
8000.001	229.104	9000.001	229.156	10000.001	229.164	12500.000	229.268	15000.001	229.614	16850.385	230.000
17500.000	230.077	20000.000	230.408	25000.000	231.123	30000.001	231.836	34795.237	232.500	40000.000	232.905
50000.000	235.657	65000.000	234.758	79999.998	260.630	50000.000	262.468	40000.000	263.500	34999.999	264.038
30000.000	264.590	20636.448	265.000	24999.999	265.300	20000.000	266.355	17500.000	266.844	14999.999	267.273
13401.642	267.500	12500.000	267.726	10000.000	268.180	9000.000	268.335	8000.000	268.512	7000.000	268.723
6000.000	268.960	5000.000	269.362	4250.778	270.000	3500.000	271.184	2500.000	274.305	64.880	300.000

***** LAYER 2 *****

** INPUT DATA **

Q= .1767124+08, UBAR AT BOTTOM= 8.0000, UBAR AT TOP= 9.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO= .0000, THETAK AT BOTTOM= 81.000
THETAK AT TOP= 82.000, Z= 194.000, ALPHA= 1.0, BETA= 1.0, DELX= .67179000+02, DELY= .26090600+03
ILMOD= 4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR = 8.50000, THETA = 81.50000, DELTHP = 1.00000, DELU = 1.00000
, SIGAP = .07854, SIGEP = .07854

***** LAYER 3 *****

** INPUT DATA **

Q= .13832881+08, UBAR AT BOTTOM= 9.0000, UBAR AT TOP= 10.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO= .0000, THETAK AT BOTTOM= 82.000
THETAK AT TOP= 82.000, Z= 250.000, ALPHA= 1.0, BETA= 1.0, DELX= .88489000+02, DELY= .26116900+03

```
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =   9.50000, THETA =  82.00000, DELTHP =    .00000, DELU =   1.00000
, SIGAP =    .07854, SIGEP =    .07854
   -----------------------------------------------------------------------------------------------------------

                                        ***** LAYER 4 *****

                                         ** INPUT DATA **

 Q=  .17571956+09, UBAR AT BOTTOM= 10.0000, UBAR AT TOP= 10.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  82.000
THETAK AT TOP=  79.000, Z=  284.000, ALPHA= 1.0, BETA= 1.0, DELX=  .27907700+03, DELY=  .26071200+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  10.00000, THETA =  80.50000, DELTHP =  -3.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   -----------------------------------------------------------------------------------------------------------

                                        ***** LAYER 5 *****

                                         ** INPUT DATA **

 Q=  .85552758+08, UBAR AT BOTTOM= 10.0000, UBAR AT TOP= 10.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  79.000
THETAK AT TOP=  79.000, Z=  500.000, ALPHA= 1.0, BETA= 1.0, DELX=  .34302300+03, DELY=  .26039300+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  10.00000, THETA =  79.00000, DELTHP =    .00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   -----------------------------------------------------------------------------------------------------------

                                        ***** LAYER 6 *****

                                         ** INPUT DATA **

 Q=  .15551787+09, UBAR AT BOTTOM= 10.0000, UBAR AT TOP= 10.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  79.000
THETAK AT TOP=  78.000, Z=  558.000, ALPHA= 1.0, BETA= 1.0, DELX=  .43896500+03, DELY=  .25997900+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  10.00000, THETA =  78.50000, DELTHP =  -1.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   -----------------------------------------------------------------------------------------------------------

                                        ***** LAYER 7 *****

                                         ** INPUT DATA **

 Q=  .33045615+09, UBAR AT BOTTOM= 10.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  78.000
THETAK AT TOP=  76.000, Z=  637.000, ALPHA= 1.0, BETA= 1.0, DELX=  .60103800+03, DELY=  .25917500+03
IZMOD=  4
```

```
CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  10.50000, THETA =  77.00000, DELTHP =  -2.00000, DELU =   1.00000
, SIGAP =    .07854, SIGEP =    .07854
   ------------------------------------------------------------------------------------------------------------

                                        ***** LAYER 8 *****

                                         ** INPUT DATA **

 Q=  .14067361+10, UBAR AT BOTTOM= 11.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  76.000
THETAK AT TOP=  71.000, Z=  750.000, ALPHA= 1.0, BETA= 1.0, DELX=  .10535510+04, DELY=  .25673400+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  11.00000, THETA =  73.50000, DELTHP =  -5.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   ------------------------------------------------------------------------------------------------------------

                                        ***** LAYER 9 *****

                                         ** INPUT DATA **

 Q=  .90119708+09, UBAR AT BOTTOM= 11.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  71.000
THETAK AT TOP=  68.000, Z= 1000.000, ALPHA= 1.0, BETA= 1.0, DELX=  .12599460+04, DELY=  .25554400+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  11.00000, THETA =  69.50000, DELTHP =  -3.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   ------------------------------------------------------------------------------------------------------------

                                        ***** LAYER10 *****

                                         ** INPUT DATA **

 Q=  .40267074+09, UBAR AT BOTTOM= 11.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  68.000
THETAK AT TOP=  67.000, Z= 1098.000, ALPHA= 1.0, BETA= 1.0, DELX=  .13407370+04, DELY=  .25505700+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  11.00000, THETA =  67.50000, DELTHP =  -1.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
   ------------------------------------------------------------------------------------------------------------

                                        ***** LAYER11 *****

                                         ** INPUT DATA **

 Q=  .14619776+10, UBAR AT BOTTOM= 11.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  67.000
THETAK AT TOP=  63.000, Z= 1135.000, ALPHA= 1.0, BETA= 1.0, DELX=  .16103340+04, DELY=  .25336000+03
IZMOD=  4

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  11.00000, THETA =  65.00000, DELTHP =  -4.00000, DELU =    .00000
```

```
, SIGAP =    .07854, SIGEP =    .07854
-----------------------------------------------------------------------------------------------------

                                   ***** LAYER12 *****

                                   ** INPUT DATA **

 Q=  .30778222+10, UBAR AT BOTTOM= 11.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000, SIGAK AT TOP= 4.50000
SIGEK AT BOTTOM= 4.50000, SIGLK AT TOP= 4.50000, SIGXO= 532.8370, SIGYO= 532.8370, SIGZO=    .0000, THETAK AT BOTTOM=  63.000
THETAK AT TOP=  56.000, Z= 1250.000, ALPHA= 1.0, BETA= 1.0, DELX=  .20839970+04, DELY=  .25016700+03
IZMOD=  4
 Z AT TOP= 1432.0000

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =  11.00000, THETA =  59.50000, DELTHP =  -7.00000, DELU =    .00000
, SIGAP =    .07854, SIGEP =    .07854
-----------------------------------------------------------------------------------------------------
```

```
*******************************************************************************
*                                                                             *
*      EXAMPLE SPACE SHUTTLE NORMAL LAUNCH                                    *
*                                                                             *
*                       KSC 21 OCT 72.                                        *
*                                                                             *
*                         DATE = 07/17/75, TIME = 16/14/18                    *
*                                                                             *
*                                                                             *
*                                                                             *
* ADJUSTED CLOUD RISE HEIGHT = 1038.20, RANGE =  4103.23, AZIMUTH BEARING = 235.45 *
*                                                                             *
*******************************************************************************
```

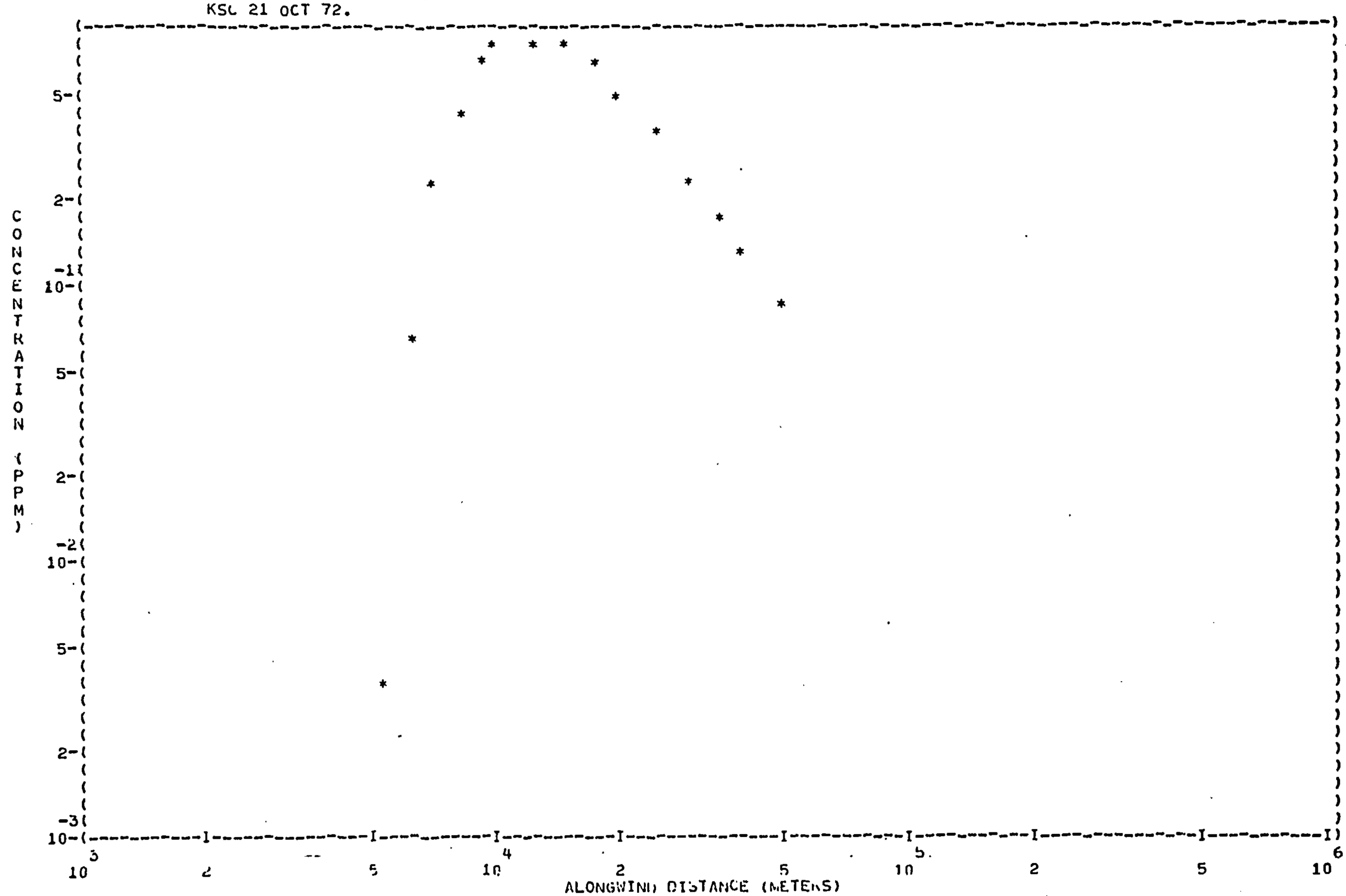
```
                                   ***** LAYER 1 *****

                                    ** INPUT DATA **

 Q=  .80917860+10, ZRR= 13.000, UBAR AT BOTTOM=  6.0000, UBAR AT TOP= 11.0000, SIGAK AT BOTTOM= 4.50000
SIGAK AT TOP= 4.50000, SIGEK AT BOTTOM= 4.50000, SIGEK AT TOP= 4.50000, TAUK= 447.273, TAUOK= 447.273
SIGXO= 532.8370, SIGYO= 532.8370, SIGZO= 183.1630, THETAK AT BOTTOM=  80.000, THETAK AT TOP=  56.000, Z=      .000
ALPHA= 1.0 BETA= 1.0, H= 1038.200, DELX=  .41032290+04, DELY=  .23545500+03, IZMOD=  3, TIM1=  .00000000
ZLIM=     .000, LAMBDA=  .0000, TIMAV= 600.000, XRY= 100.000, YRZ= 100.000, XLPY=     .000, XLRZ=     .000, GAMMAP= .000
 Z AT TOP= 1452.0000

CALCULATED INPUT PARAMETERS FOR MODELS 1,2,3 **** UBAR =   9.71775, THETA =  68.00000, DELTHP = -24.00000, DELU =   5.00000
, SIGAP =    .07854, SIGEP =    .07854
       -----------------------------------------------------------------------------------------------------------
```

MAXIMUM CENTERLINE HCL CALCULATIONS AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

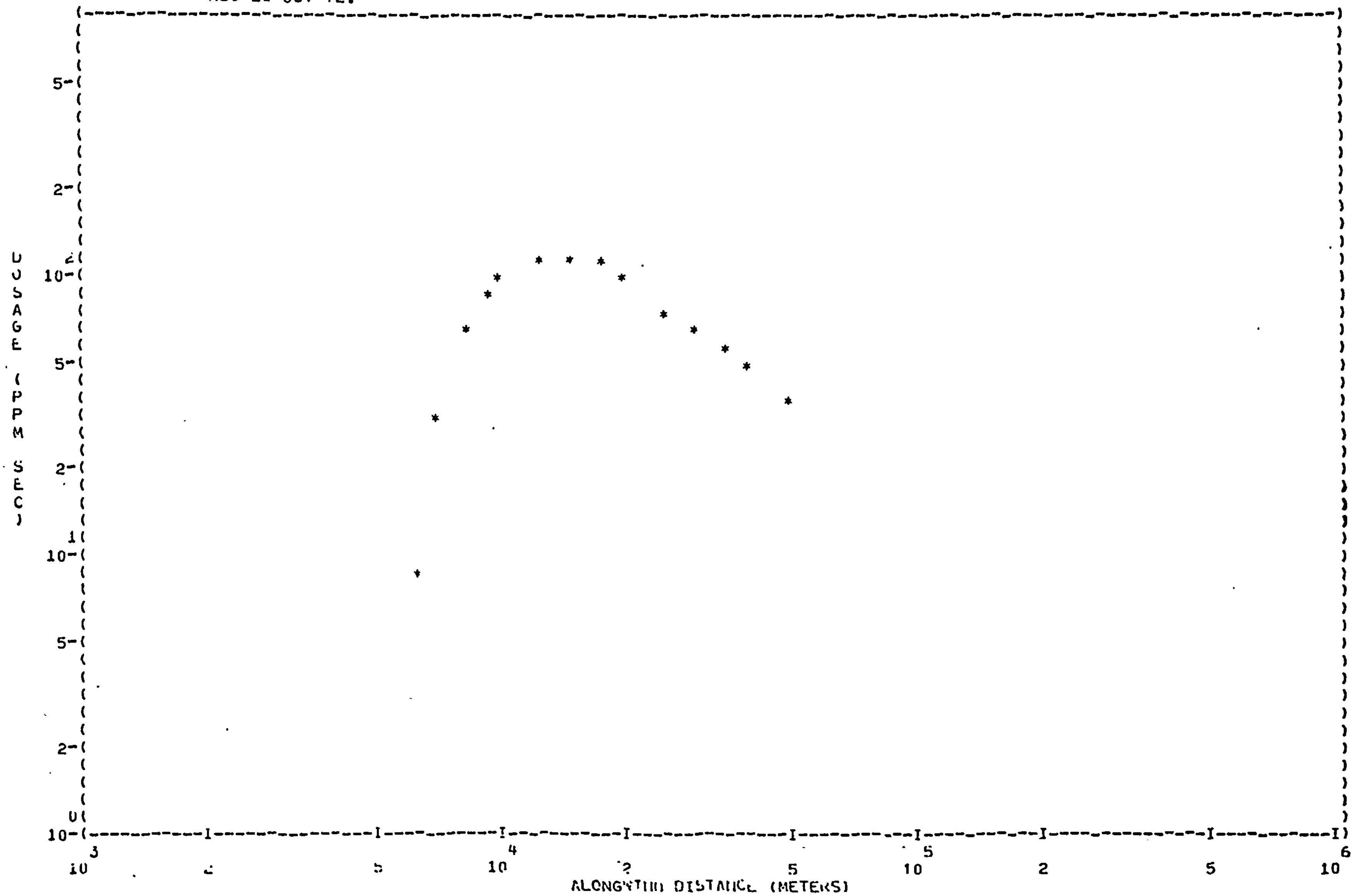
.788 IS THE MAXIMUM PEAK CONCENTRATION
122.735 IS THE MAXIMUM DOSAGE
.205 IS THE MAXIMUM PEAK 10.0 MINUTE TIME-MEAN CONCENTRATION

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	MAXIMUM PEAK CONCENTRATION (PPM)	MAXIMUM DOSAGE (PPM SEC)	MAXIMUM PEAK 10.0 MINUTE TIME-MEAN CONCENTRATION (PPM)	TIME OF CLOUD PASSAGE (SECONDS)	AVERAGE CLOUD CONCENTRATION (PPM)
5000.0	240.4	.003	.470	.001	236.196	.002
6000.0	240.6	.067	9.218	.015	237.531	.039
7000.0	241.2	.243	33.956	.057	239.784	.142
8000.0	241.9	.451	63.878	.106	242.936	.263
9000.0	242.5	.615	88.541	.148	246.946	.359
10000.0	243.0	.719	105.552	.176	251.770	.419
12500.0	243.9	.788	122.735	.205	267.077	.460
15000.0	244.6	.717	119.633	.199	286.410	.418
17500.0	245.0	.605	108.895	.181	308.985	.352
20000.0	245.4	.498	96.999	.162	334.174	.290
25000.0	245.9	.341	77.579	.129	390.269	.199
30000.0	246.3	.244	64.253	.107	451.609	.142
35000.0	246.5	.182	54.807	.090	516.298	.106
40000.0	246.7	.141	47.774	.077	583.250	.082
50000.0	246.9	.090	38.006	.059	721.357	.053

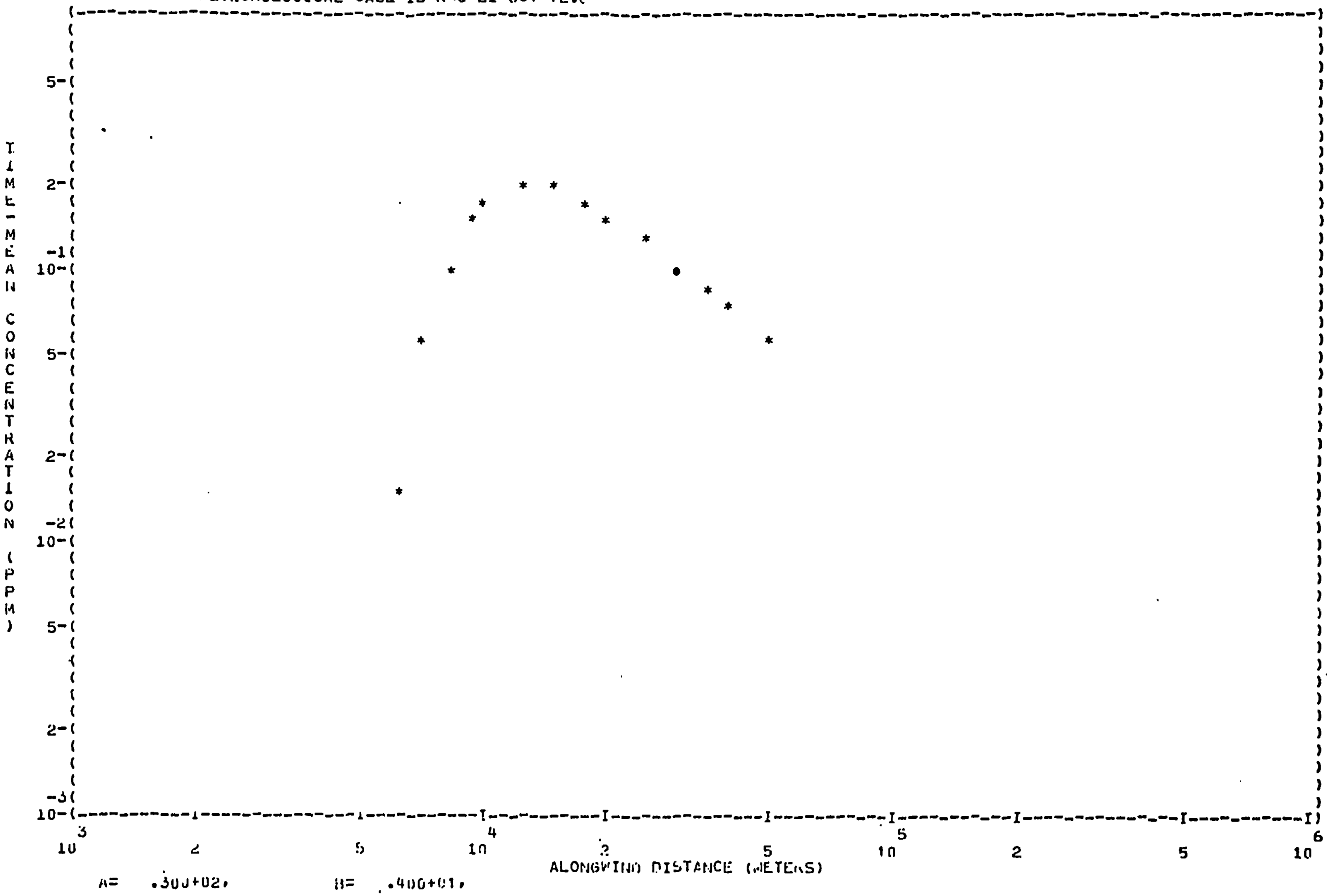
MAXIMUM CENTERLINE HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



MAXIMUM CENTERLINE HCL DOSAGE IN PPM SEC AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



MAXIMUM CENTERLINE HCL 10 MINUTE TIME-MEAN CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.



ISOPLETHS HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

--* ISOPLETH LEVEL = .500, CONCENTRATION (PPM)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
8479.742	240.000	9000.000	238.377	9824.584	237.500	11502.030	237.500	12500.000	237.582	15000.000	238.897
16583.469	240.000	17500.000	240.965	18960.599	242.500	19934.082	245.000	19314.856	247.500	17500.000	249.111
15825.102	250.000	15000.000	250.281	12500.000	250.289	12046.868	250.000	10000.000	248.563	9376.508	247.500
9000.000	246.581	6578.366	245.000	8306.618	242.500						

--* ISOPLETH LEVEL = .100, CONCENTRATION (PPM)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
6195.893	240.000	6522.493	235.000	8000.001	230.711	9000.001	230.219	10000.001	230.159	12500.001	230.563
15000.000	231.319	17500.000	232.367	17812.657	232.500	20000.000	233.224	24580.074	235.000	25000.000	235.183
30000.000	236.974	31560.847	237.500	35000.000	238.847	38137.632	240.000	40000.000	240.944	44077.667	242.500
47385.351	245.000	46002.933	247.500	45950.121	250.000	40000.000	252.494	35000.000	254.170	32234.965	255.000
30000.000	255.581	25000.000	256.765	20473.890	257.500	20000.000	257.587	17500.000	257.904	15000.000	257.907
13278.011	257.500	12500.000	257.387	10000.000	256.160	9065.872	255.000	8000.001	253.306	7024.326	250.000
6310.196	245.000										

ISOPLETHS HCL DOSAGE IN PPM SEC AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

--* ISOPLETH LEVEL = 100.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
10000.000	241.173	11499.040	240.000	12500.000	239.704	13443.331	240.000	15000.000	240.672	17500.000	242.307
19342.969	245.003	16042.315	247.500	17500.000	247.777	15000.000	248.494	12500.000	248.128	11806.448	247.500
10000.000	245.102	9684.162	242.500								

--* ISOPLETH LEVEL = 50.000, DOSAGE (PPM SEC)

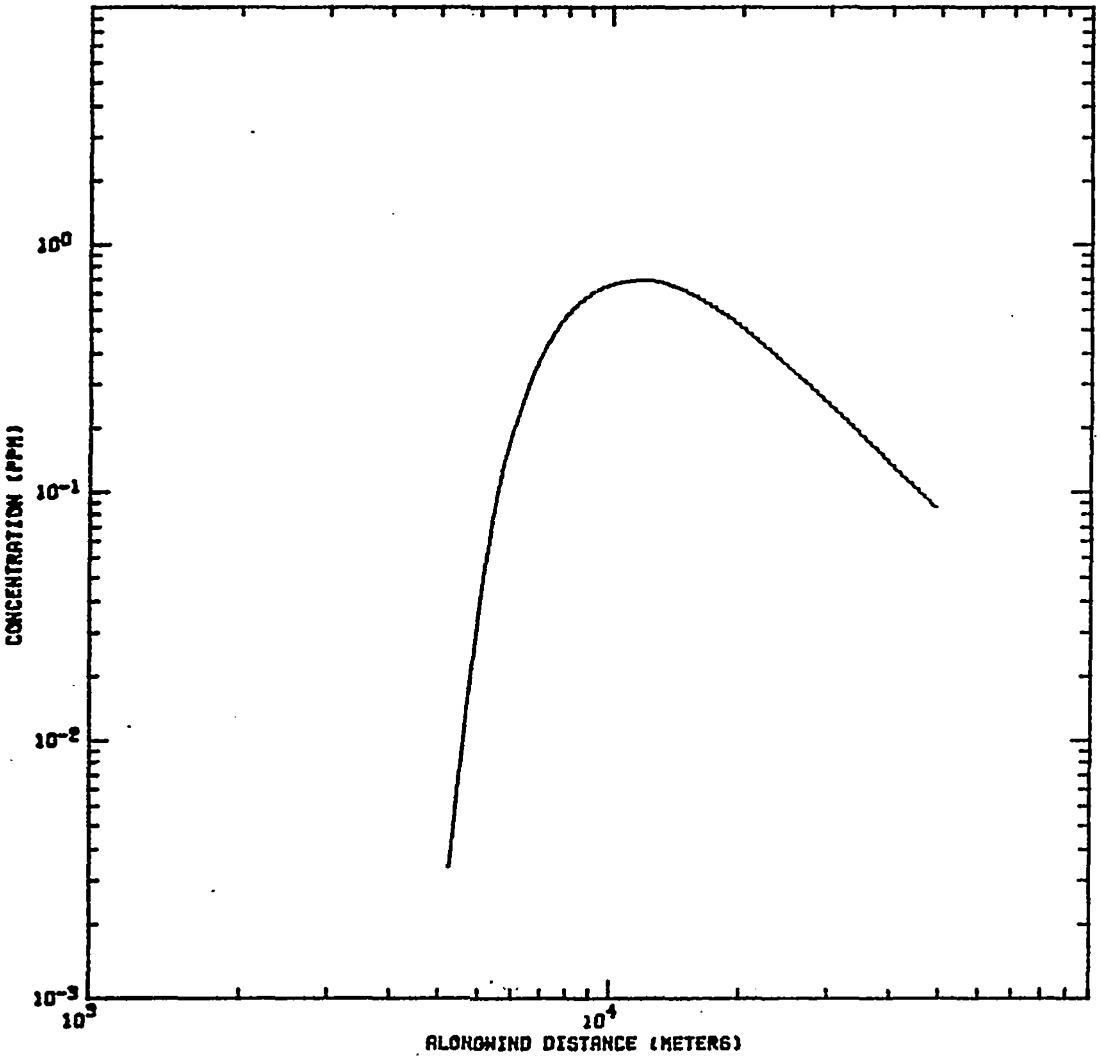
RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
7965.976	237.500	9000.001	235.542	10000.000	235.015	12500.001	235.100	15000.001	235.746	17500.000	236.620
19725.666	237.500	20000.000	237.614	25000.000	239.499	26582.323	240.000	30000.000	241.447	33007.956	242.500
35000.000	243.760	37473.169	245.000	38143.867	247.500	35000.000	249.321	30000.000	251.110	25000.000	252.376
24462.403	252.500	20000.000	253.295	17500.000	253.551	15000.000	253.512	12500.000	252.883	11964.034	252.500
10000.000	251.082	9000.001	249.492	8264.691	247.500	7550.387	242.500				

--* ISOPLETH LEVEL = 25.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
7264.799	235.000	8663.306	232.500	12500.001	232.233	15000.001	232.841	17500.001	233.498	20000.001	234.251
22668.566	235.000	25000.001	235.626	30000.000	236.778	33398.294	237.500	35000.000	237.843	40000.000	238.767
47368.076	240.000	61725.603	242.500	72499.601	245.000	76031.759	247.500	70775.987	250.000	57299.939	252.500
49999.999	253.380	40000.000	254.645	37018.661	255.000	34999.999	255.237	29999.999	255.821	24999.999	256.328
19999.999	256.682	17499.999	256.718	15000.000	256.534	12500.000	255.871	11236.577	255.000	9000.000	252.956
7160.766	247.500	6655.104	240.000								

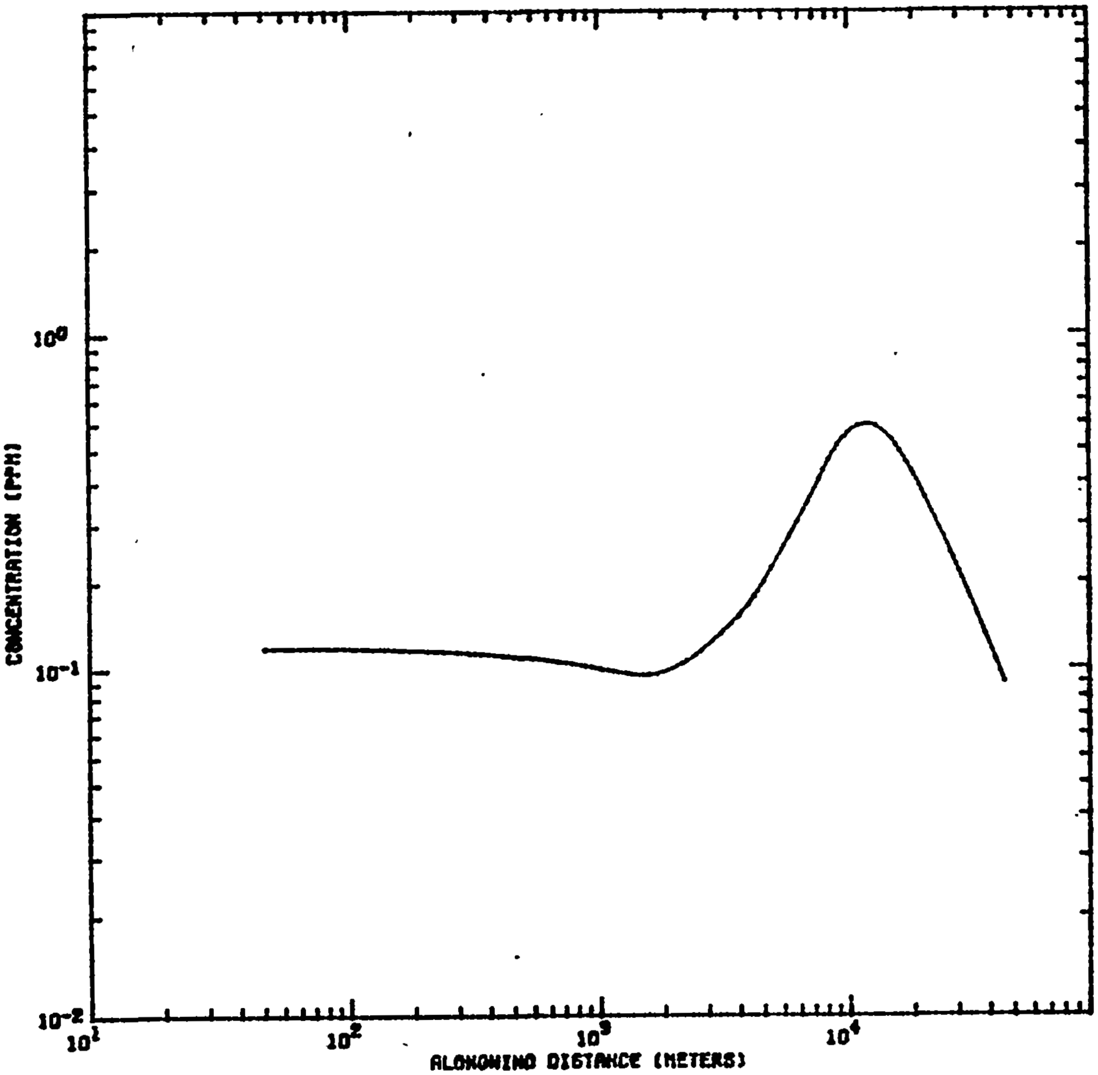
--* ISOPLETH LEVEL = 5.000, DOSAGE (PPM SEC)

RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)	RANGE (METERS)	AZIMUTH BEARING (DEGREES)
5799.881	235.000	6603.690	230.000	8000.001	227.631	9000.001	227.250	10000.001	227.226	11211.726	227.500
12500.001	227.642	15000.001	228.049	17500.001	228.553	20000.000	229.134	23712.746	230.000	25000.001	230.202
30000.001	230.836	35000.000	231.468	40000.000	232.100	43797.968	232.500	50000.000	232.980	65000.000	234.024
79283.634	235.000	79999.999	259.735	64999.999	260.352	49999.999	261.065	39999.999	261.488	34999.999	261.670
29999.999	261.821	24999.998	261.918	19999.999	261.899	17500.000	261.778	15000.000	261.466	12500.000	260.681
11498.758	260.000	10000.000	259.260	9000.000	258.207	8000.000	256.699	7000.001	254.031	6179.228	250.000
5543.363	242.500										



MAXIMUM CENTERLINE HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

FIGURE D-1. Maximum centerline HCL concentration for Model 3 generated by the plot routines for the NASA/MSFC Multilayer Diffusion Models Program.



MAXIMUM CENTERLINE HCL CONCENTRATION IN PPM AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 4 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

FIGURE D-2. Maximum centerline HCL concentration from Model 4 generated by the plot routines for the NASA/MSFC Multilayer Diffusion Models Program.

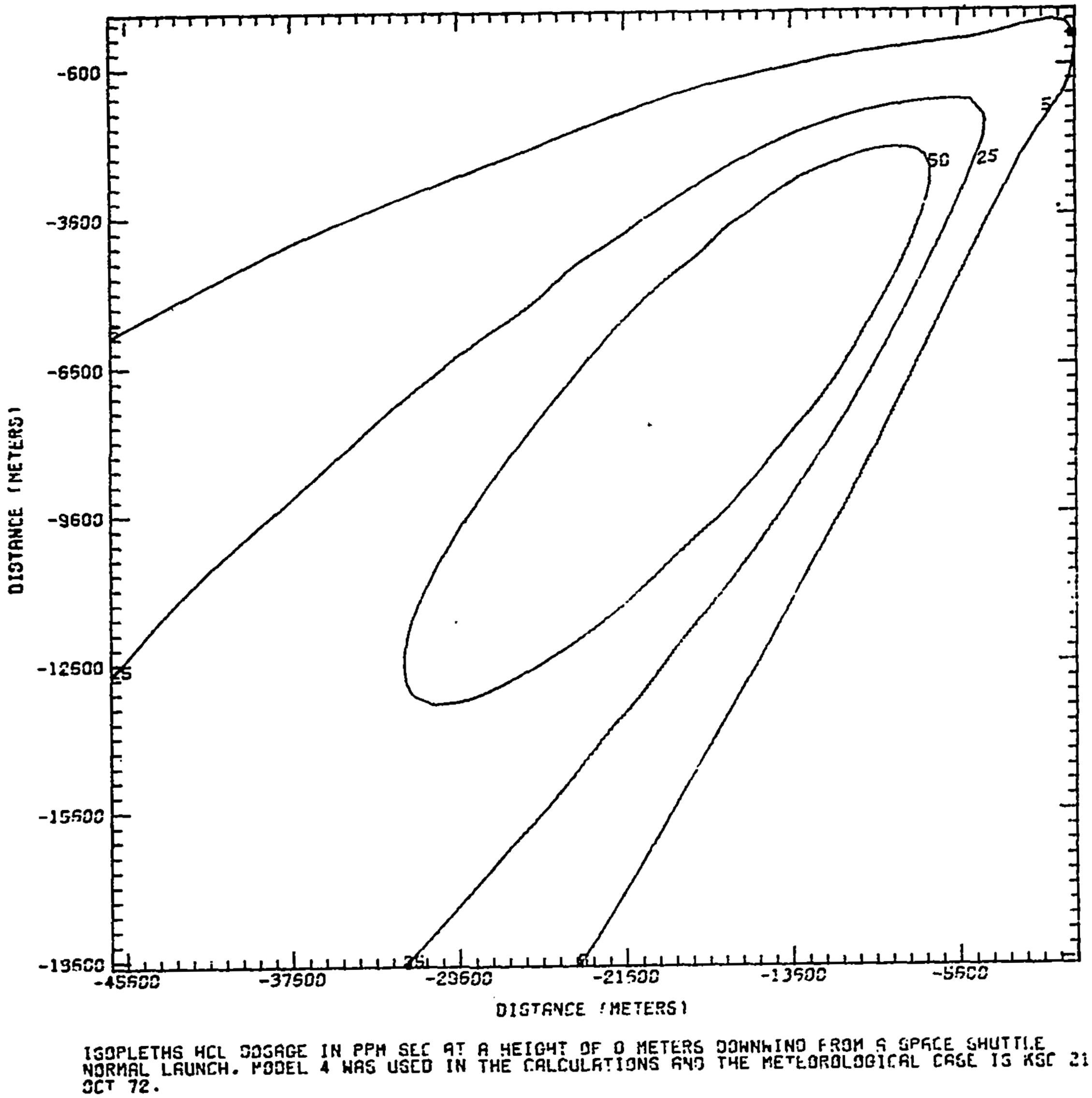
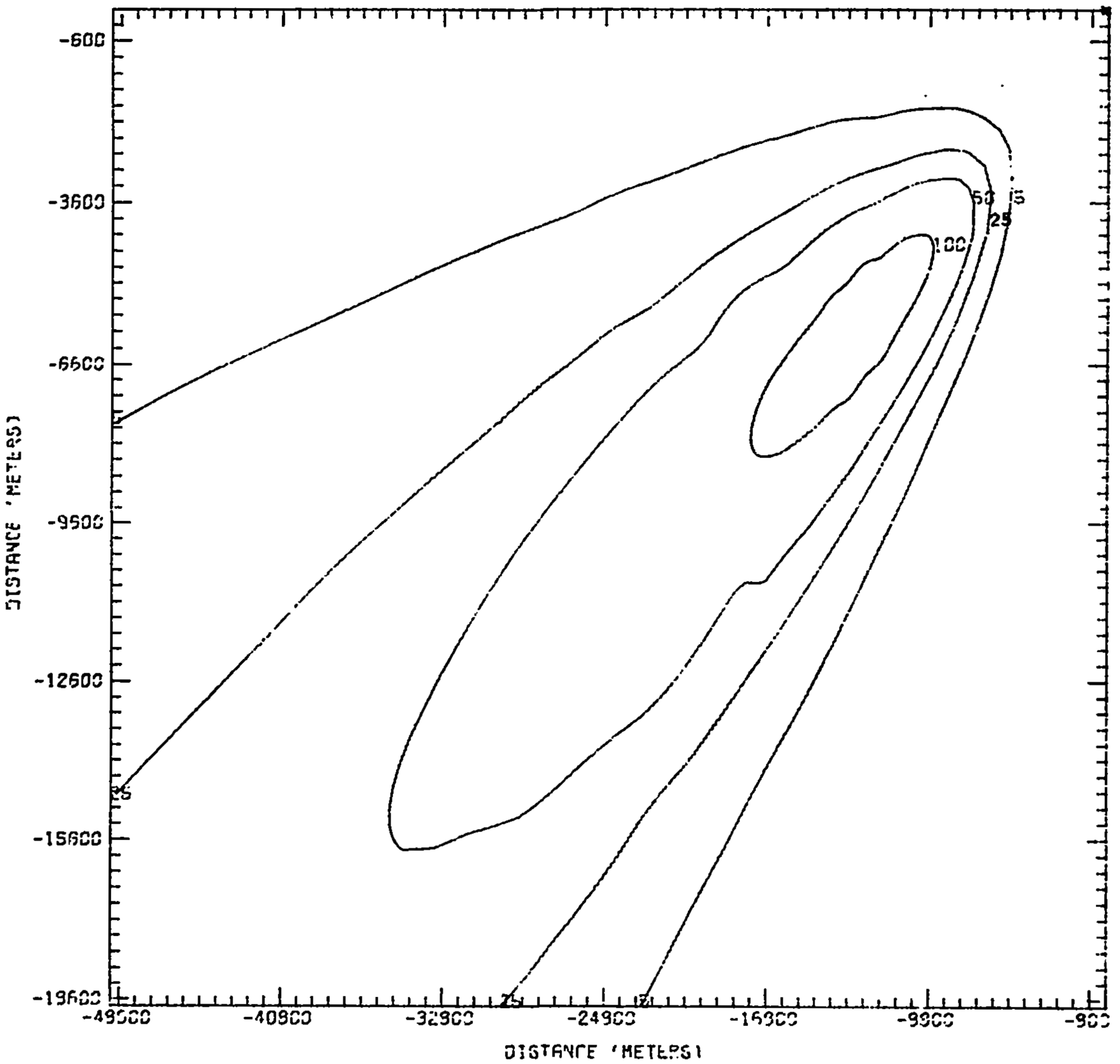


FIGURE D-3. Isopleths of HCL dosage for Model 3 generated by the plot routines for the NASA/MSFC Multilayer Diffusion Models Program.



ISOPLETHS HCL DOSAGE IN PPM SEC AT A HEIGHT OF 0 METERS DOWNWIND FROM A SPACE SHUTTLE NORMAL LAUNCH. MODEL 3 WAS USED IN THE CALCULATIONS AND THE METEOROLOGICAL CASE IS KSC 21 OCT 72.

FIGURE D-4. Isopleths of HCL dosage for Model 4 generaged by the plot routines for the NASA/MSFC Multilayer Diffusion Models Program.

APPENDIX E

DERIVATION OF THE VERTICAL TERM APPEARING IN EQUATION (3-18), SECTION 3

The Vertical Term defined by the summation in Equation (3-18), Section 3 of the main body of this report was derived using the method of image sources (Slade, 1968, p. 346) to account for the reflection of gases and aerosols at the earth's surface and at the bases of elevated inversions. Consider a point source located at a height H above the surface (z = 0) within a surface mixing layer of depth H_m, as shown in Figure E-1. The numbers appearing at the height z in Figure E-1 denote the intersections of rays from the real and image sources with the height z at various distances downwind from the source. Thus, the number 1 indicates the ray intersection from the real source at height H with the level z and the number 4 indicates the intersection of the ray from the image source at height $(2H_m + H - z)$. The heights of other image sources above the z plane are indicated at the left of Figure E-1. The parameter γ_r is the fraction of material reflected at the earth's surface. For complete reflection, γ_r is set equal to unity while, for no reflection, γ_r is zero. Note that the height term for each image source is multiplied by γ_r for each ray intersection with the earth's surface that occurs prior to the intersection of the ray with height z. Therefore, the image source at the height $(2H_m + H - z)$ is multiplied by γ_r for the reflection at the earth's surface from the image source at $(H + z)$; the image source at $(2H_m + H + z)$ is multiplied by γ_r^2 for the reflection at the earth's surface from the image sources at $(H + z)$ and $(2H_m + H + z)$. The contributions to the vertical term in Equation (3-18) from the real and image sources, assuming a Gaussian distribution of material in the vertical, are seen from an inspection of Figure E-1 to be given by the following expressions:

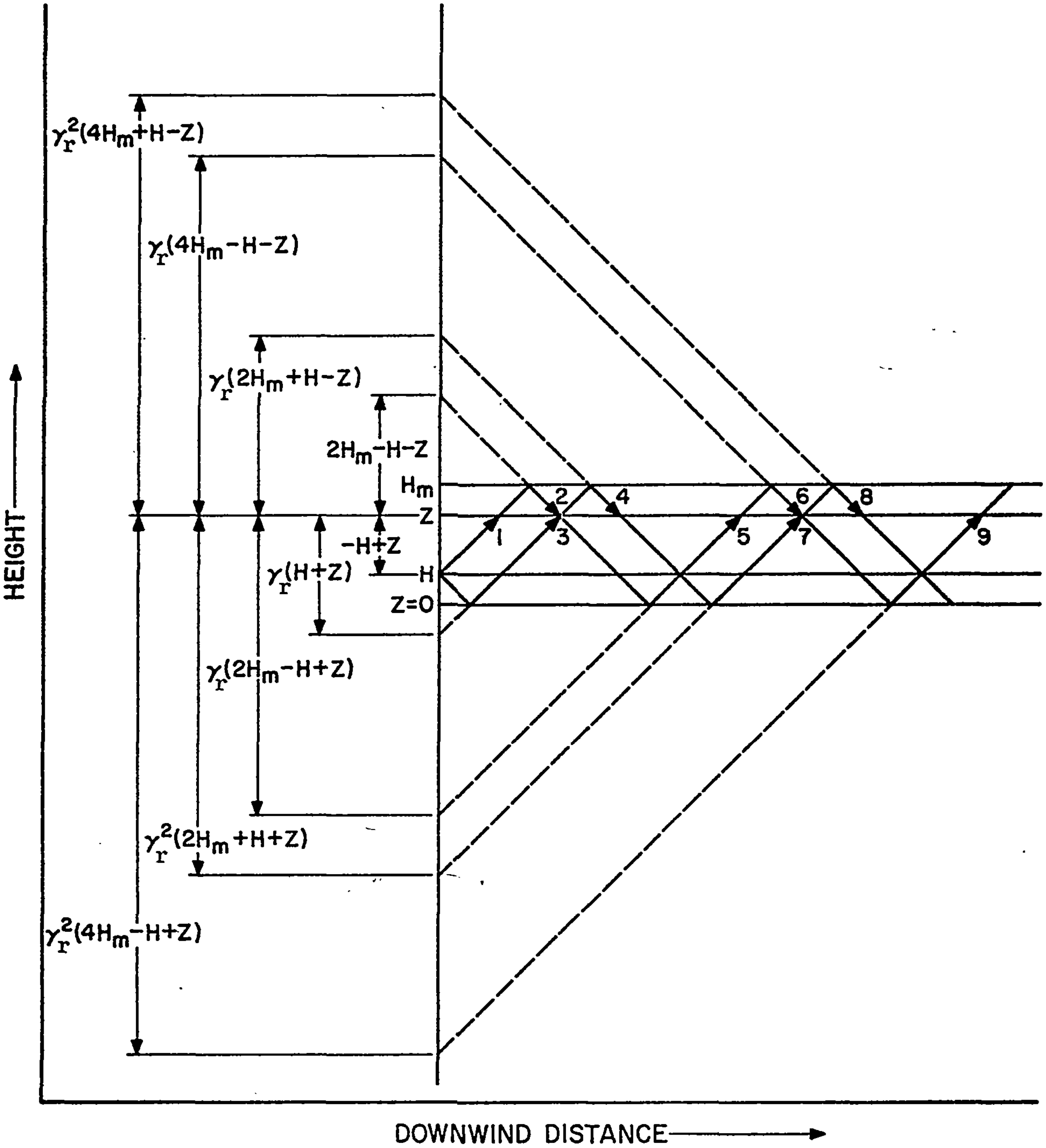


FIGURE E-1. Schematic diagram of image source configurations.

Contribution at Point Number	Term
1	$\exp\left\{\frac{-(H-z)^2}{2\sigma_z^2}\right\}$
2	$\exp\left\{\frac{-\left(2H_m - H - z\right)^2}{2\sigma_z^2}\right\}$
3	$\gamma_r \exp\left\{\frac{-(H+z)^2}{2\sigma_z^2}\right\}$
4	$\gamma_r \exp\left\{\frac{-\left(2H_m + H - z\right)^2}{2\sigma_z^2}\right\}$
5	$\gamma_r \exp\left\{\frac{-\left(2H_m - H + z\right)^2}{2\sigma_z^2}\right\}$
6	$\gamma_r \exp\left\{\frac{-\left(4H_m - H - z\right)^2}{2\sigma_z^2}\right\}$
7	$\gamma_r^2 \exp\left\{\frac{-\left(2H_m + H + z\right)^2}{2\sigma_z^2}\right\}$
8	$\gamma_r^2 \exp\left\{\frac{-\left(4H_m + H - z\right)^2}{2\sigma_z^2}\right\}$

Contribution at Point Number	Term
9	$\gamma_r^2 \exp\left\{\dfrac{-\left(4H_m - H + z\right)^2}{2\sigma_z^2}\right\}$
$\vdots$	
i	

If we sum over all sources and combine terms, the total Vertical Term becomes

$$
\begin{aligned}
&\sum_{i=0}^{\infty} \left\{ \gamma_r^i \exp\left[-1/2\left(\frac{2iH_m + H - z}{\sigma_z}\right)^2\right] + \gamma_r^{i+1} \exp\left[-1/2\left(\frac{2iH_m + H + z}{\sigma_z}\right)^2\right]\right\} \\
+ &\sum_{i=1}^{\infty} \left\{ \gamma_r^i \exp\left[-1/2\left(\frac{2iH_m - H + z}{\sigma_z}\right)^2\right] + \gamma_r^{i-1} \exp\left[-1/2\left(\frac{2iH_m - H - z}{\sigma_z}\right)^2\right]\right\}
\end{aligned}
\qquad \text{(E-1}
$$

where, for convenience in writing Equation (E-1), the quantity 0^0 is defined to be unity.

Next, consider a vertical line source of length L centered about height H in Figure E-1. Assuming that the line source is comprised of an infinite number of point sources, the Vertical Term for the real line source V_H centered at height H is given by the expression

$$
V_H = \frac{1}{\sqrt{2\pi}\,\sigma_z} \int_{H-L/2}^{H+L/2} \exp\left\{\frac{-(z' - z)^2}{2\sigma_z^2}\right\} dz' \qquad \text{(E-2)}
$$

If the substitution

$$\xi = \frac{z' - z}{\sqrt{2}\,\sigma_z}$$

is made in Equation (E-2), the resulting expression is

$$V_H = \frac{1}{\sqrt{\pi}} \int_{\frac{-L/2+(H-z)}{\sqrt{2}\,\sigma_z}}^{\frac{L/2+(H-z)}{\sqrt{2}\,\sigma_z}} \exp\left(-\xi^2\right) d\xi \tag{E-3}$$

Using the definition of error functions (Abramowitz and Stegun, 1964, p. 297), Equation (E-3) can be written in the form

$$\begin{aligned} V_H &= \frac{1}{2}\left\{\mathrm{erf}\left(\frac{L/2 + (H-z)}{\sqrt{2}\,\sigma_z}\right) - \mathrm{erf}\left(\frac{-L/2 + (H-z)}{\sqrt{2}\,\sigma_z}\right)\right\} \\ &= \frac{1}{2}\left\{\mathrm{erf}\left(\frac{L/2 + (H-z)}{\sqrt{2}\,\sigma_z}\right) + \mathrm{erf}\left(\frac{L/2 - (H-z)}{\sqrt{2}\,\sigma_z}\right)\right\} \end{aligned} \tag{E-4}$$

Inspection of Euation (E-4) shows that the numerators of the error function arguments express, respectively, the distance of the top and base of the line source from the height z. The corresponding vertical terms for point and line sources in the surface mixing layer, including provision for partial reflection at the earth's surface, are given in Table E-1.

The Vertical Term of Equation (3-18) can be obtained from the line source terms in Table E-1 through use of the following relationships:

$$2H_m = 2\left(z_{TL} - z_{BL}\right) \tag{E-5}$$

TABLE E-1

CORRESPONDING VERTICAL TERMS OF A POINT SOURCE AND LINE SOURCE IN THE SURFACE MIXING LAYER

Point Source	Line Source
$\exp\left[-\frac{1}{2}\left(\frac{H-z}{\sigma_z}\right)^2\right]$	$\frac{1}{2}\left\{\mathrm{erf}\left(\frac{L/2+(H-z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(H-z)}{\sqrt{2}\,\sigma_z}\right)\right\}$
$\gamma_r\exp\left[-\frac{1}{2}\left(\frac{H+z}{\sigma_z}\right)^2\right]$	$\frac{\gamma_r}{2}\left\{\mathrm{erf}\left(\frac{L/2+(H+z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(H+z)}{\sqrt{2}\,\sigma_z}\right)\right\}$
$\exp\left[-\frac{1}{2}\left(\frac{2iH_m-H-z}{\sigma_z}\right)^2\right]$	$\frac{1}{2}\left\{\mathrm{erf}\left(\frac{L/2+(2iH_m-H-z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(2iH_m-H-z)}{\sqrt{2}\,\sigma_z}\right)\right\}$
$\gamma_r\exp\left[-\frac{1}{2}\left(\frac{2iH_m-H+z}{\sigma_z}\right)^2\right]$	$\frac{\gamma_r}{2}\left\{\mathrm{erf}\left(\frac{L/2+(2iH_m-H+z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(2iH_m-H+z)}{\sqrt{2}\,\sigma_z}\right)\right\}$
$\gamma_r\exp\left[-\frac{1}{2}\left(\frac{2iH_m+H-z}{\sigma_z}\right)^2\right]$	$\frac{\gamma_r}{2}\left\{\mathrm{erf}\left(\frac{L/2+(2iH_m+H-z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(2iH_m+H-z)}{\sqrt{2}\,\sigma_z}\right)\right\}$
$\gamma_r^2\exp\left[-\frac{1}{2}\left(\frac{2iH_m+H+z}{\sigma_z}\right)^2\right]$	$\frac{\gamma_r^2}{2}\left\{\mathrm{erf}\left(\frac{L/2+(2iH_m+H+z)}{\sqrt{2}\,\sigma_z}\right)+\mathrm{erf}\left(\frac{L/2-(2iH_m+H+z)}{\sqrt{2}\,\sigma_z}\right)\right\}$

$$H = \left(\frac{z_{TK} - z_{BK}}{2} + z_{BK} - z_{BL} \right) \tag{E-6}$$

$$z = \left(z_L - z_{BL} \right) \tag{E-7}$$

$$\frac{L}{2} = \frac{z_{TK} - z_{BK}}{2} \tag{E-8}$$

where Equations (E-5) through (E-8) express the relationship between the height coordinate system of Figure E-1 and the generalized height coordinate system used in developing Equation (3-18).

Substituting Equations (E-5) through (E-8) into the line source terms in the right-hand column of Table E-1 and collecting terms results in the expression

$$\begin{aligned}
1/2 \Bigg\{ \sum_{i=0}^{\infty} \Bigg[\gamma_r^i \Bigg[&\mathrm{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) + z_{TK} - z_L}{\sqrt{2}\ \sigma_{zLK}} \right) + \mathrm{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) - z_{BK} + z_L}{\sqrt{2}\ \sigma_{zLK}} \right) \Bigg] \\
+ \gamma_r^{i+1} \Bigg[&\mathrm{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) - 2z_{BL} + z_{TK} + z_L}{\sqrt{2}\ \sigma_{zLK}} \right) + \mathrm{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) + 2z_{BL} - z_{BK} - z_L}{\sqrt{2}\ \sigma_{zLK}} \right) \Bigg] \Bigg] \\
+ \gamma_r^{i-1} \Bigg[&\mathrm{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) + 2z_{BL} - z_{BK} - z_L}{\sqrt{2}\ \sigma_{zLK}} \right) + \mathrm{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) - 2z_{BL} + z_{TK} + z_L}{\sqrt{2}\ \sigma_{zLK}} \right) \Bigg] \Bigg] \Bigg\} \\
+ \sum_{i=1}^{\infty} \Bigg[\gamma_r^i \Bigg[&\mathrm{erf}\left(\frac{2i\left(z_{TL} - z_{BL}\right) - z_{BK} + z_L}{\sqrt{2}\ \sigma_{zLK}} \right) + \mathrm{erf}\left(\frac{-2i\left(z_{TL} - z_{BL}\right) + z_{TK} - z_L}{\sqrt{2}\ \sigma_{zLK}} \right) \Bigg] \Bigg]
\end{aligned} \tag{E-9}$$

*U.S. GOVERNMENT PRINTING OFFICE: 1975 - 635-275/48

where, again 0^0 is defined equal to unity. In Equation (3-18), the factor 1/2 appearing before the bracket in Equation (E-9) is contained in the form factor $1/2\sqrt{2\pi}$ appearing in the denominator of the first term on the right following the equal sign.