(NASA-CR-144301) SOLID ROCKET BOOSTER THERMAL RADIATION MODEL. VOLUME 2: USER'S MANUAL Final Report (Lockheed Missiles and Space Co.) 121 p HC $5.50 CSCL 21H

N76-24351

Unclas
G3/20 28106

Lockheed

HUNTSVILLE RESEARCH & ENGINEERING CENTER

LOCKHEED MISSILES & SPACE COMPANY, INC.
A SUBSIDIARY OF LOCKHEED AIRCRAFT CORPORATION

HUNTSVILLE, ALABAMA

Lockheed
Missiles & Space Company, Inc.
HUNTSVILLE RESEARCH & ENGINEERING CENTER
Cummings Research Park
4800 Bradford Drive,
Huntsville, Alabama

SOLID ROCKET BOOSTER THERMAL RADIATION MODEL – VOLUME II

USER'S MANUAL

March 1976

Contract NAS8-31310

Prepared for National Aeronautics and Space Administration
Marshall Space Flight Center, Alabama 35812

by

A. L. Lee

APPROVED: B. Hobson Shirley
B. Hobson Shirley, Supervisor
Engineering Sciences Section

J. S. Farrior
J. S. Farrior
Resident Director

FOREWORD

This user's manual is a supplement to "Solid Rocket Booster Thermal Radiation Model – Volume I – Final Report," LMSC-HREC TR D496763-I. This manual was prepared by personnel of the Thermal & Fluid Physics Group, Engineering Sciences Section, of the Lockheed-Huntsville Research & Engineering Center under Contract NAS8-31310. The contract period of performance was from 20 January 1975 through 20 March 1976. The work was administered under the technical direction of Mr. William C. Claunch of the Structures and Propulsion Laboratory, NASA-Marshall Space Flight Center.

CONTENTS

1. STRUCTURE OF THE PROGRAM

The SRB plume thermal radiation program is stored on a magnetic tape which was created by a Univac 1108 7-track tape drive. There are two entry points corresponding to the two main programs on the tape. The main program, MAINS, deals with a single plume, which was used during the phases of development and checkout and for making the data tapes. The second main program, MAIN, computes the heating rate due to dual plume configuration and view factor calculation. Many subroutines on the tape are common to both MAINS and MAIN programs. In its logical structure, the MAINS program is the same as the MAIN program, minus the ICALC = 2 option, which calculates the view factors. The logical structure of MAIN program is shown in Chart 1.

The entire code consists of 2 main programs, 24 subroutines, 1 PROC and 3 elements. The relations between the main programs and the subroutines are summarized in Table 1. The PROC defines dimension statements which are used in the COMMON blocks. They are inserted in the program by using an INCLUDE statement. The elements are used to list the entire program, to compile the program elements when the array sizes are changed and to punch the program deck. More about these elements is discussed in Section 4. where the run characteristics are concerned.

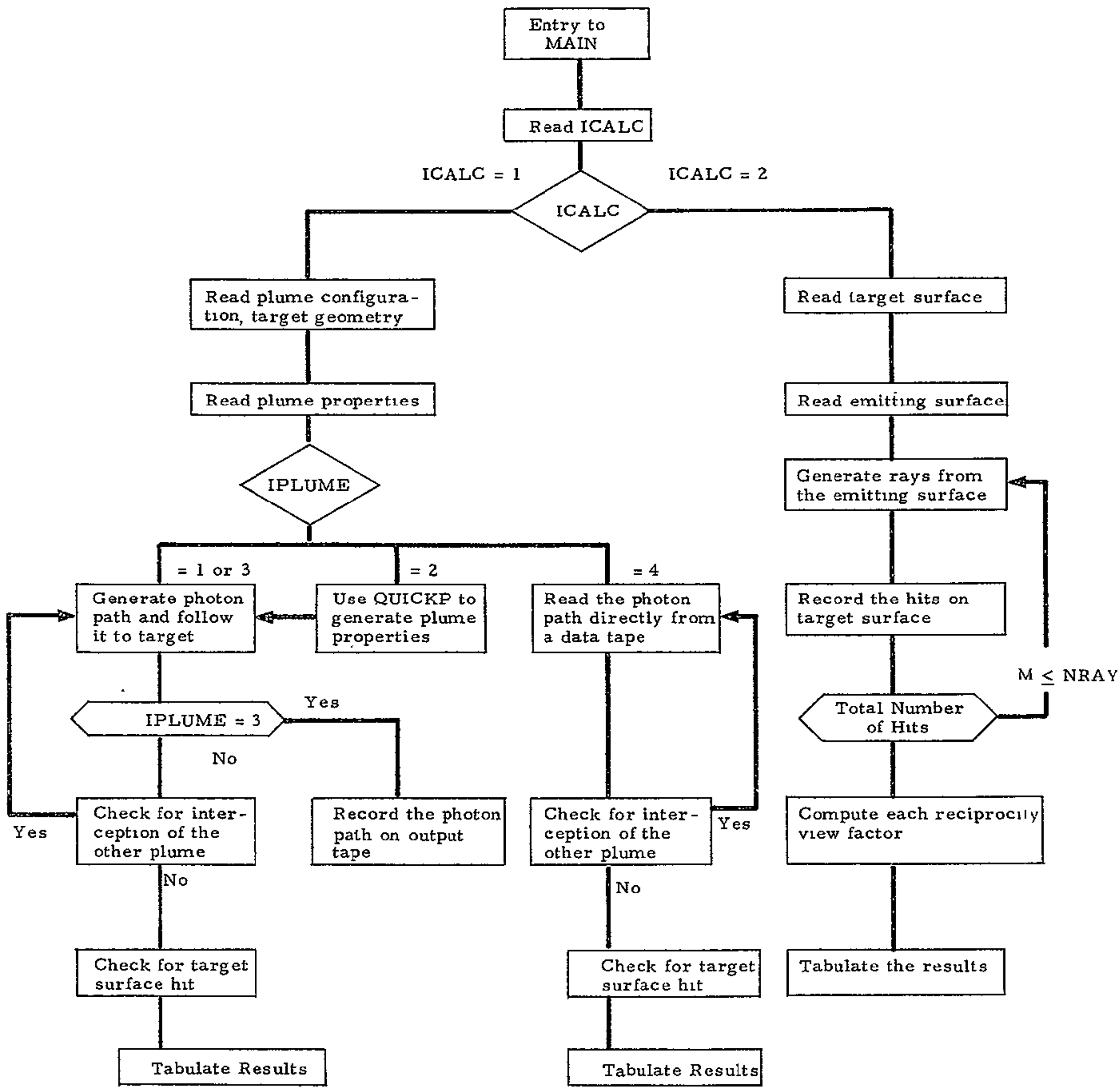

Chart 1 - The Logical Structure of the MAIN Program

Table 1

RELATIONS BETWEEN MAIN PROGRAMS AND SUBROUTINES

Subroutine	Single Plume (MAINS) Heating Rate	Dual Plume (MAIN)	
		Heating Rate ICALC = 1	View Factor ICALC = 2
DIMENS (PROC)	x	x	x
ATTEN	x	x	
CHOSE	x	x	x
DIFVDC			x
DISK	x	x	x
EMITT	x	x	
ESCAP		x	
ESCAPE	x		
FRTAPE	x	x	
INPUT	x	x	
INTRCP		x	
IOPKT	x	x	
OUTPUT	x	x	
PINGEA	x	x	x
PINGEB	x	x	x
QUADEQ	x	x	x
QUICKP	x	x	
SCATTR	x	x	
SORTNG	x	x	x
SPHERE	x	x	
TARGET	x	x	x
TRANSF		x	x
VFEMIT			x
VFOUTP			x
ZCOORD	x	x	

"x" indicates the requirement of the subroutine.

2. INPUT GUIDE

The input cards for the program can be organized into five groups. Each group of input cards is read by a program element, i.e., either the main program or a subroutine, except as noted. Table 2 summarizes the input card groups.

Table 2
INPUT CARD GROUPS

Input Card Group	Incurred by	Single Plume Heating Rate	Dual Plume Heating Rate	Dual Plume View Factor
1	MAIN		x	x
2	TARGET	x	x	x
3	INPUT	x	x	
4	FRTAPE	x	x	
5*	VFEMIT			x

*The first card reading NRAY and NSTART is requested in the MAIN program.
"x" indicates the requirement of the data card group.

As is apparent in Table 2, not all the five groups of input cards are required in a data card ensemble. The single plume heating rate calculation, for example, requires the input cards only from groups 2, 3 and 4. Preparation of input cards for each group will be discussed in detail in the following paragraphs.

2.1 INPUT CARD GROUP 1

This group is required in the MAIN program and consists of two cards:

Card 1: (I8) ICALC

Card 2: (10F8.0) ((PP(I, J), J = 1, 3), SIG(I), PSI(I), I = 1, 2)

ICALC indicates if this run is for view factor calculation in which case ICALC = 2 or for heating rate calculation in which case ICALC can be any value other than 2. It is suggested that ICALC = 1 be used to indicate heating rate calculation. When the view factor calculation is intended (where ICALC = 2), the card 2 is omitted. The input format is indicated in parentheses.

The card 2 reads the basic coordinate systems of the dual plumes. (PP(1, 1), PP(1, 2), PP(1, 3)) are the (X1, X2, X3) coordinates of the center of the exit plane of the first plume with respect to the central coordinate system. SIG(1) and PSI(1) is the σ and ψ angles of the axis of the plume. (PP(2, 1), PP(2, 2), PP(2, 3)), SIG(2), PSI(2) are the corresponding values for the second plume. The coordinate system is shown in Fig. 1. The central coordinate system is centered at the mid point between the two ungimbaled plume exit planes. The coordinate systems (X1', X2', X3') and (X1", X2", X3") are the local coordinates aligned with the first and second plume, respectively. All distances are non-dimensionalized with respect to nozzle radius at the exit plane.

2.2 INPUT CARD GROUP 2

This group, which is read by calling TARGET subroutine in all calculation cases, describes the target geometry. There is really no limit as to how many target surfaces can be considered in the calculation. However, in the present setup of the program array dimensions, the number of target surfaces is not to exceed 10. Each target surface is described by a set of two cards, described as follows:

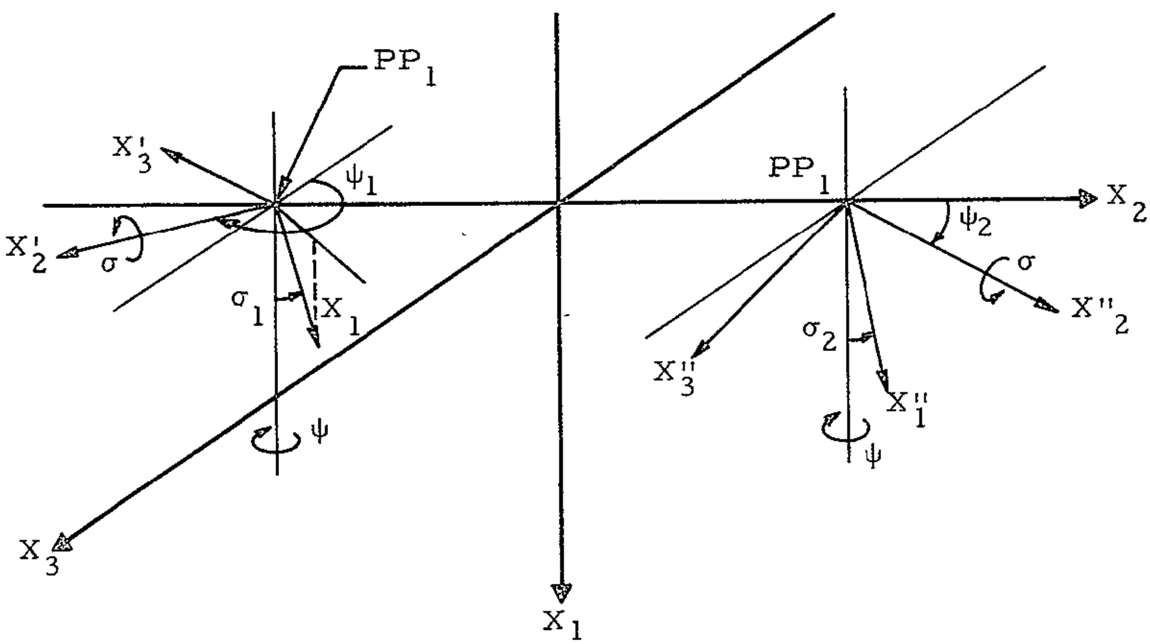

Fig 2-1 - Coordinate Systems of Gimbaled Dual Plumes

Card 1: (5I4, 4X, 7F8.0) IPTION, NN1, NN2, NRING1, NRING2, RADISK, RCP1, RCP2, RADISU

Card 2: (9F8.0) X10, X20, X30, X40, X50, X60, X70, X80, X90

This group of cards is ended by adding a blank card following the second card of the last set. All lengths are in non-dimensional units.

IPTION,	target surface code IPTION = 1, Cylinder; 2, Frustum; 3, Paraboloid; 4, Ellipsoid, 5, Parallelogram; 6, Annular Disk
NN1	number of view points along arc length, or along P1-P2 line in parallelogram case
NN2	number of view points along axial direction, or along P2-P3 line in parallelogram case, or along the radial direction in disk case
NRING1	number of view point areas along radial direction on the constraint disk passing through the point P1. This is applicable to cylinder and frustum only.
NRING2	Same as above except for the constraint plane passing through the point P2. This is applicable to cylinder, frustum and paraboloid only.
RADISK	radius of the constraint disk passing the point P2 for the cases of cylinder, frustum and paraboloid; the outer radius in the case of annular disk; not applicable in the cases of ellipsoid or parallelogram
RCP1	radius of the inner radius on the constraint disk passing through point P1. This is applicable to cylinder and frustum. In the case of annular disk, RCP1 is the inner radius of the disk.
RCP2	radius of the inner radius on the constraint disk passing through point P2. This is applicable to cylinder, frustum and paraboloid.
RADISU:	radius of the constraint disk passing through P1 for the case of frustum only
X10, X20, X30	the (X1, X2, X3) coordinates of point P1
X40, X50, X60	the coordinates of P2
X70, X80, X90	the coordinates of P3, except the annular disk case where (X70, X80, X90) are the outward normal vector components

The designations of P1, P2 and P3 are summarized in Table 3.

Table 3

DESIGNATIONS OF POINTS P_1, P_2 AND P_3

<table>
<tr><th>IPTION Shape</th><th>P_1
(X10, X20, X30)</th><th>P_2
(X40, X50, X60)</th><th>P_3
(X70, X80, X90)</th></tr>
<tr><td>1
Cylinder</td><td rowspan="2">Center of the Top</td><td rowspan="3">Center of the Base</td><td rowspan="3">An Arbitrary Point not on the Axis</td></tr>
<tr><td>2
Frustum</td></tr>
<tr><td>3
Paraboloid</td><td>The Vertex</td></tr>
<tr><td>4
Ellipsoid</td><td>Center of the Body</td><td>The Pole</td><td>At the Zero Meridian</td></tr>
<tr><td>5
Parallelogram</td><td colspan="3">The points P_1, P_2, P_3 are the three consecutive corners of the plane, clockwise on the plane, looking in the direction of the negative normal.</td></tr>
<tr><td>6
Annular Disk</td><td>Center of the Disk</td><td>On the rim of the disk. P_1-P_2 forms the line from which the view vectors count in right-hand rule sense with its normal.</td><td>P_3 represents the unit normal vector of the disk</td></tr>
</table>

2.3 INPUT CARD GROUP 3

This group is read by INPUT subroutine. The first card of this group carries some control parameters.

Card 1: (7I8, F8.0) IX, JX, IRGN, JRGN, NSTART, ISO, IPLUME, REX

IX, JX	printout control parameters. The results of heating rate computations will be printed (IX) times at (JX) sample increments, i.e., IX = 3, JX = 2000, the results will be printed out 3 times when 2000, 4000 and 6000 samples are generated, respectively.
IRGN, JRGN	number of regions the plume body is divided in longitudinal and radial directions, respectively
NSTART	a starting random number, any six-digit integer
ISO	use 0. to indicate if isotropic scattering is desired; use 1 to indicate anisotropic scattering. In latter case, cards 7 and 8 of this group are required.
IPLUME	run options
= 1	to compute the heating rate on target surfaces. card 6 in this group is omitted in this case.
= 2	to compute the heating rate on target surface. The plume is defined by QUICKP subroutine. cards 2, 3, 4 and 5 are omitted in this case.
= 3	interception of target surface is not tested (therefore the Input Card Group 2 needs only a blank card); trajectories of the photons are recorded in a data tape (when this option is used, a cataloged tape must be assigned to Unit 10). The output tape can then be used with IPLUME = 4 option.
= 4	the trajectories of the photons are read from an input data tape which must be assigned to Unit 10 at the start of the run. In this case, cards 2 through 8 are omitted.
REX	the radius of the exit plane in physical units (cm). For the standard SRB REX = 185 cm (6.07 ft).

Card 2: (F10.0) GAMMA

Card 2 is repeated (JRGN) times. Each GAMMA is the half cone angle of the concentric conic division within the plume.

Card 3: (F10.0) HZ(I)

Card 4: (6F10.0) (PROP(K, I, J), K = 1,6)

Card 5: (6F10.0) (PROP(K, I, J), K = 7,10)

HZ(I) is the longitudinal division of the plume body. PROP(K, I, J) is the K^{th} property of the plume in the I^{th} longitudinal division and J^{th} radial division. The index K defines the plume property as shown in Table 4.

Table 4
DEFINITION OF PROP(K, I, J) ARRAY

K	Plume Property
1	$A\ell_2O_3$ Particle Number Density, N (parts/ft^3)
2	$A\ell_2O_3$ Particle Temperature, T_p (R)
3	$A\ell_2O_3$ Particle Radius, r_p (ft)
4	$\sum N r_p^2$ (parts/ft)
5	Gas Temperature, T_g (R)
6	Gas Pressure, P_g (lb/ft^2)
7	Mole Fraction for CO
8	Mole Fraction for CO_2
9	Mole Fraction for H_2O
10	Mole Fraction for HCl

Cards 3, 4 and 5 are in a loop and are repeated (IRGN + 1) times; cards 4 and 5 are in an inner loop, repeating (JRGN + 1) times.

Cards 2 to 5 are put together as a package as the result of a plume flow field computation.

Card 6: (4F10.0) PC, PAMB, TC, XK

This card is used only when IPLUME = 2

PC	pressure in the combustion chamber (lb/ft^2)
PAMB	ambient pressure (lb/ft^2)
TC	combustion chamber temperature (R)
XK	polytropic exponent

Card 7: (5F10.0) SA, SB, SC, SD, SE

Card 8: (5F10.0) SF, SG, SH, SI, SJ

Cards 7 and 8 are required only when anisotropic scattering option (ISO = 1) is used. The quantities SA, SB, ... etc., define linear segments of the scattering distribution curve.

As an example, a set of the values of SA, SB, etc., are given below.

SA	SB	SC	SD	SE	SF	SG	SH	SI	SJ
4.0	15.0	80.0	160.0	175.0	0.2	4.0	15.0	4.2	160.0

2.4 INPUT CARD GROUP 4

This group consists of only one data card and is read in FRTAPE subroutine only when IPLUM = 4 option is used.

Card 1: (3I8, 4F8.0) KEY, ITG, NTRAJ, DELPHI, TX, TY, TZ

KEY	control parameter
= 1	compute the heating rate for the entire geometry ensemble

= 2	for a single target surrounding the plumes, in this case DELPHI = 2π
= 3	single small target within DELPHI
= 4	single small target, using cosine projection to compute the heating rate
= 0	terminate the run
= -1	to tabulate the distribution of the sample trajectories on the data tape in increments of ϕ, η, θ and X1
ITG	the identity of the target surface of which the heating rate is to be computed. This parameter is used in KEY = 3 and 4 cases.
NTRAJ	number of the trajectories to be read from the data tape
DELPHI	the angle subtended by the target surface as viewed from the origin point of the coordinate system. This parameter is used in KEY = 3 and 4 cases.
TX, TY, TZ	the center location of the target surface or the intersection point of the centerline of the DELPHI cone and the target surface. These points are used in KEY = 3 and 4 cases.

2.5 INPUT CARD GROUP 5

This group consists of 3 cards.

Card 1: (2I8) NRAY, NSTART

NRAY	number of sample sizes used to calculate the view factor
NSTART	a starting random number, any six-digit integer

Card 2: (I8) IEMIT

IEMIT	the identifying code of the emitting surface
= 1	half cylinder
= 2	half frustum
= 3	not used
= 4	hemisphere
= 5	parallelogram
= 6	annular disk

= 7	full cylinder
= 8	full frustum
= 9	sphere

Card 3: (10F8.0)

This card reads 10 non-dimensional quantities to define the emitting surface. They are summarized in Table 5.

Table 5

DESCRIPTION OF THE EMITTING SURFACES

IEMIT	R_1	P_1	P_2	P_3 or DC
1 Half Cylinder	Radius of the cylinder	Center of the top	Center of the bottom	Normal at mid arc
2 Half Frustum	Radius of the top	Center of the top	Center of the bottom	Mid arc point of the bottom
3 Not Used				
4 Hemisphere	Radius of the sphere	Center of the sphere	Normal at the center of the surface	
5 Parallelogram		P_1, P_2, P_3 are three consecutive corners of the plane in counterclockwise direction around the normal.		
6 Annular Disk	Inner radius of the disk	Center of the disk	A point on the outer periphery	Normal of the disk
7 Full Cylinder	Radius of the cylinder	Center of the top	Center of the bottom	Normal at an arbitrary point on the surface
8 Full Frustum	Radius of the sphere	Center of the top	Center of the bottom	A point on the periphery on the bottom
9 Sphere	Radius of the sphere	Center of the sphere		

3. EXAMPLES OF INPUT CARDS AND OUTPUT PRINTOUT

Three complete run decks listed on pages 16 through 21 show the input cards.

The first deck is an IPLUME = 1 case for the dual plume heating rate computation. The complete deck that defines a sea level plume with afterburning is included in the listing. This deck is generated from the Lockheed plume flowfield program. The listing actually shows the particle number density, N, (PROP(1, I, J)) in units of $P/ft^3 \times 10^{-9}$. In the IPLUME = 1, 2 and 3 cases, the INPUT subroutine, where the Input Card Group 3 is read, is called again to input the next case provided there are no changes in the target geometries. A blank card following last input card will terminate the run.

The second deck is an IPLUME = 4 case for the single plume heating rate computation where an input data tape assigned to unit 10 is used. The last data card can be repeated as many times as the case may be. A blank card following the last data card terminates the run.

The third deck is an example of calculating the view factors. The data cards read as Input Card Group 5 can be repeated as many times as desired to compute the view factors on a set of target surfaces from different emitting surfaces. A blank card following the last data card terminates the run.

```
'RUN RADIAT,1HNI3V451053,LEE-ALBIN202,19,150
'ASG,T TAPE1,T,13215
'REWIND TAPE1
'COPY,G TAPE1,TPF$
'FREE TAPE1
'SETC,I
'MAP,IS
 IN MAIN
 LIB SYS$*MSFC$.
'XQT
       1
 .0     -3.4315 0.0      0.0       0.0       0.0      3.4315    0.0     0.0      0.0
   5   12    8    0    0
 .0     -6.0     -4.0  -  0.0       6.0      -4.0      0.0      6.0     4.0

       9     5000       14        5   123456        0         1  185.0
 1.2
 2.4
 3.6
 4.6
 6.0
   .0
  .2571        4078.    .196       .0988      3729.      1302.
  .1603       .2483     .0275      .1537
  .27         4077.     .1957      .1034      3752.      1400.
  .1604       .2484     .0274      .1537
  .2971       4081.     .1952      .1132      3802.      1629.
  .1606       .2487     .027       .1539
  .31         4085.     .195       .12        3849.      1918.
  .1609       .249      .0267      .1538
  .39         4080.     .196       .144       3883.      2175.
  .1612       .2492     .0265      .1538
  .2675       3371.     .1033      .03        3523.      1835.
  .16         .2469     .0297      .1508
  .4
  .2571       4077.9    .196       .0988      3729.      1302.
  .1603       .2483     .0275      .1537
  .27         4076.     .196       .1034      3752.      1400.
  .1604       .2484     .0274      .1536
  .297        4082.     .195       .113       3802.      1629.
  .1606       .2488     .0268      .1538
  .31         4085.     .195       .12        3849.      1918.
  .1609       .249      .0267      .1538
  .39         4080.     .196       .144       4000.      2028.
  .1612       .2492     .0265      .1538
  .268        3371.     .103       .03        1522.      2028.
  .035        0.001     .02        .0096
  .8
  .2571       4078.     .196       .0988      3814.      1364.
  .1604       .2486     .0269      .1543
  .27         4076.     .196       .1024      3804.      1391.
  .1604       .2486     .027       .1542
  .297        4082.     .195       .113       3804.      1492.
  .1605       .2487     .027       .154
  .31         4085.     .195       .12        3812.      1634.
  .1606       .2487     .027       .1539
  .39         4080.     .196       .144       4079.      2029.
  .1469       .225      .0231      .1518
```

.268	3371.	.103	.03	4000.	2029.
.12	.1	.04	.08		
1.2					
.257	4078.	.196	.0988	3873.	1416.
.1605	.2489	.0265	.1547		
.27	4076.	.196	.1034	3851.	1411.
.1605	.2488	.0266	.1546		
.297	4082.	.16	.155	3804.	1395.
.1605	.2487	.0269	.1544		
.31	4085.	.195	.12	3697.	1268.
.1602	.2482	.0277	.1538		
.39	4080.	.196	.144	4127.	0229.
.1554	.2157	.0251	.1469		
.268	3371.	.103	.03	4000.	2029.
.18	.1	.05	.08		
2.13					
.257	4078.	.196	.0988	3834.	1272.
.1604	.2487	.0267	.1548		
.27	4076.	.196	.1034	3775.	1195.
.1602	.2485	.0272	.1546		
.297	4082.	.195	.113	3685.	1080.
.16	.248	.0279	.154		
.31	4085.	.195	.12	3700.	1130.
.1601	.248	.0275	.1541		
.37	4082.	.196	.14	3909.	1626.
.1607	.249	.0262	.1547		
.2	3370.	.06	.01	4179.	2029.
.1746	.007	.0989	.0562		
2.49					
.23	4070.	.196	.098	3718.	1034.
.16	.2482	.0275	.1546		
.25	4075.	.196	.105	3686.	1011.
.16	.248	.0278	.1543		
.275	4080.	.196	.12	3685.	1041.
.1599	.248	.0278	.1542		
.33	4080.	.195	.14	3822.	1348.
.1604	.2487	.0268	.1546		
.32	4080.	.194	.135	4000.	1800.
.18	.2	.04	.13		
.09	4000.	.1	.07	4200.	2029.
.18	.01	.1	.06		
2.69					
.22	4060.	.196	.097	3611.	849.
.1594	.2474	.0286	.1537		
.23	4080.	.196	.1	3636.	909.
.1596	.2476	.028	.1541		
.26	4080.	.196	.11	3770.	1170.
.1601	.2484	.0273	.1546		
.315	4080.	.195	.13	3930.	1600.
.1607	.2492	.026	.1549		
.305	4080.	.194	.125	4323.	2029.
.2130	.1221	.0554	.1057		
.05	3500.	.1	.03	4000.	2029.
.2	.01	.09	.06		
3.22					
.21	3900.	.190	.075	3581.	785.
.1592	.247	.0289	.1534		
.24	3975.	.195	.1	3700.	976.

.16	.248	.0277	.1545		
.29	4000.	.195	.11	3870.	1471.
.1606	.249	.0262	.1549		
.31	4030.	.196	.115	4000.	1824.
.1609	.2495	.0255	.1551		
.3	3970.	.1	.135	4340.	2029.
.2139	.119	.0568	.1043		
.1	3400.	.1	.02	4000.	2029.
.1899	.0153	.1015	.0632		
3.6					
.21	3850.	.19	.07	3775.	1044.
.1601	.2484	.0271	.1549		
.235	3900.	.195	.09	3920.	1400.
.1605	.249	.026	.155		
.28	4000.	.195	.1	4003.	1730.
.1608	.2494	.0256	.1551		
.3095	4050.	.195	.11	4055.	1940.
.161	.25	.0252	.1552		
.27	3850.	.19	.12	4357.	2029.
.2151	.114	.06	.1024		
.05	3500.	.1	.005	2858.	2029.
.0905	.0001	.05	.03		
4.27					
.21	3900.	.185	.072	4304.	2371.
.1607	.2501	.0238	.1549		
.25	3940.	.185	.09	4130.	1975.
.1611	.25	.0246	.1555		
.27	4000.	.186	.1	4080.	1990.
.161	.2498	.025	.1552		
.295	3900.	.186	.1	4150.	2050.
.18	.2	.032	.135		
.001	4000.	.15	.0001	4370.	2029.
.21	.05	.08	.075		
.00001	4000.	.05	.000001	3156.	2029.
.1055	.00002	.063	.0367		
4.89					
.46	4219.	.179	.15	4444.	3544.
.16	.2501	.0235	.1539		
.445	4177.	.181	.147	4385.	3300.
.1607	.2502	.0236	.1543		
.39	4175.	.185	.133	4209.	2469.
.1613	.2503	.0242	.1553		
.05	4000.	.18	.1	4190.	2029.
.1997	.1534	.044	.12		
.000001	3000.	.01	.000001	4000.	2029.
.15	.005	.085	.05		
.000001	3000.	.01	.000001	3000.	2029.
.08	.00001	.04	.025		
6.49					
.4961	4264.	.185	.17	4260.	2973.
.1612	.2502	.0242	.1545		
.48	4200.	.166	.168	4197.	2931.
.1616	.2503	.0244	.1548		
.385	4175.	.166	.133	4183.	2692.
.1616	.2503	.0243	.1553		
.234	3800.	.191	.085	4435.	2029.
.2182	.0886	.0709	.0923		
.000001	4000.	.01	.000001	3681.	2029.

.1347	.0006	.08	.0444		
.000001	3000.	.01	.000001	2503.	2029.
.0352	.00001	.0069	.0004		
8.23					
.37	4171.	.191	.135	4011.	1686.
.1612	.2497	.0255	.1549		
.35	4171.	.191	.126	3979.	1869.
.161	.2497	.0257	.1548		
.32	4129.	.189	.113	4010.	1895.
.1609	.2496	.0255	.155		
.235	3900.	.1895	.085	4400.	2029.
.215	.1	.07	.09		
.000001	3000.	.01	.000001	3900.	2029.
.15	.005	.083	.05		
.000001	3000.	.01	.000001	2700.	2029.
.07	.00001	.04	.025		
9.19					
.35	4171.	.19	.133	3945.	1493.
.1607	.2492	.026	.1549		
.33	4170.	.19	.128	3945.	1564.
.1607	.2492	.026	.155		
.31	4130.	.19	.155	4100.	2000.
.2025	.1481	.0453	.1168		
.22	4000.	.18	.085	4466.	2029.
.2173	.0723	.079	.086		
.000001	3000.	.01	.000001	3800.	2029.
.12	.001	.063	.04		
.000001	3000.	.01	.000001	2182.	2029.
.059	.00001	.0353	.021		
12.3					
.253	4152.	.188	.09	4167.	2029.
.1655	.2038	.0279	.141		
.265	4050.	.188	.092	4167.	2029.
.182	.18	.035	.13		
.2	4000.	.19	.072	4320.	2029.
.2125	.1238	.0547	.1064		
.000001	4000.	.01	.000001	4200.	2029.
.18	.02	.085	.06		
.000001	3500.	.01	.000001	3500.	2029.
.12	.00001	.065	.044		
.000001	2000.	.01	.000001	2000.	2029.
.06	.00001	.03	.02		

'FIN

```
'RUN RADIAT,1HNTSV451053,LEE-ALBIN202,5,100
'ASG,T TAPE1,T,14445
'ASG,T 10,T,09052  . SPSL DATA
'REWIND TAPE1
'COPY,G TAPE1,TPF$
'FREE TAPE1
'SETC,I
'MAP,IS
 IN MAINS
 LIB SYS$*MSFC$.
'XQT
   2    4    4    1    4    1.0     0.0     0.0     0.5
 1.5    0.0     0.0     0.0     0.0     0.0     0.0     1.0     0.0
   1    4    4    1    1    0.5     0.0     0.0     0.5
 2.5    0.0     0.0     -1.5    0.0     0.0     -1.5    0.5     0.0
   2    8   10    4    4    3.0     0.5     2.5     2.25
 2.5    0.0     0.0     -0.5    0.0     0.0     -0.5    3.0     0.0
   6    8    4    0    0    2.625   2.125
 1.5    0.0     0.0     -1.5    2.625   0.0     1.0     0.0     0.0

                                       123456       0       4 185.0
        1         1   50000

'FIN
```

Input Example 3

```
'RUN   RADIAT,1HNTSV451053,LEE-ALBIN202,9,150
'ASG,T TAPE1,T,13328
'REWIND TAPE1
'COPY,G TAPE1,TPF$
'FREE  TAPE1
'MAP,IS
 IN MAIN
 LIB SYS$*MSFC$.
'XQT
       2
   2    8    3    2    2     1.01      0.0       0.0       2.303
 .0      -3.4315 0.0       12.3      -3.4315  0.0       0.0       -2.4315  0.0
   2    8.   3    2    2     1.01      0.0       0.0       2.303
 .0       3.4315 0.0       12.3       3.4315  0.0       0.0        4.4315  0.0
   6    2   20    1    1     10.0      2.0
 .0       0.0       0.0       0.0       10.0      0.0       1.0       0.0       0.0

    10000   987654
       3
1.0       0.0       0.0       0.0       1.0      70.0       0.0
    10000   987654
       5
 .0      -1.0       0.0       0.0       1.0       0.0       0.0       1.0       2.0
    10000   987654
       6
 .0       0.0       3.4315    0.0       0.0       4.4315    0.0       1.0       0.0       0.0
    10000   987654
       1
2.0       0.0       0.0      -5.0       4.0       0.0      -5.0       0.0       0.0       1.0
    10000   987654
       2
2.0       0.0       0.0       5.0       4.0       0.0       5.0       4.0       0.0       4.0

'FIN
```

The output of the program consists of four main parts, which correspond to output examples 1 to 4, respectively.

The first part of the output is the description of the input targer geometry. Part of the printout is shown as the output example 1 (page 23). The first two lines print out the input read by the TARGET subroutine. The transformation matrix and the coefficients of the quadric equation follow. Then the view point vectors from the origin of the central coordinate system to the center of the area segments on the target surface are printed out. The numbers in numerical order on the left column are the index numbers for the subareas, which are used throughout the output.

Output Example 1

```
 INPUT DATA FOR TARGET NO.  1

 2    4    3    2    2        2.0000  1.0000    .0000  1.0000
 .0000   .0000   .0000 -1.0000  -.0000  .0000 -1.0000  2.0000   .0000

TRANSFORMATION MATRIX
   -.100000+01        .000000          -.198419-08
    .000000          .100000+01        -.000000
    .198419-08        .000000          -.100000+01

COEFFICIENTS FOR A CONE
C1  =   -.400000+01
C2  =    .100000+01
C3  =    .100000+01
C4  =    .000000
C5  =   -.198419-07
C6  =    .000000
C7  =    .000000
C8  =    .000000
C9  =    .000000
CONST =     .000000
RADIUS OF CONTRAINT DISK =   .200000+01

COEFFICIENTS FOR CONSTRAINT PLANES
C11  =   -.100000+01
C12  =    .000000
C13  =   -.198419-08
CEND   =    .000000
CBASE=    .100000+01
REFERENCE POINTS ON CONSTRAINT DISK     T(  -1.00000   2.00000    .00000 )

COMPONENTS FOR VIEW POINT VECTORS
     1.    VP1= -.833333+00    VP2=  .117851+01    VP3=  .117851+01
     2.    VP1= -.833333+00    VP2= -.117851+01    VP3=  .117851+01
     3.    VP1= -.833333+00    VP2= -.117851+01    VP3= -.117851+01
     4.    VP1= -.833333+00    VP2=  .117851+01    VP3= -.117851+01
     5.    VP1= -.500000+00    VP2=  .707107+00    VP3=  .707107+00
     6.    VP1= -.500000+00    VP2= -.707107+00    VP3=  .707107+00
     7.    VP1= -.500000+00    VP2= -.707107+00    VP3= -.707107+00
     8.    VP1= -.500000+00    VP2=  .707107+00    VP3= -.707107+00
     9.    VP1= -.166667+00    VP2=  .235702+00    VP3=  .235702+00
    10.    VP1= -.166667+00    VP2= -.235702+00    VP3=  .235702+00
    11.    VP1= -.166667+00    VP2= -.235702+00    VP3= -.235702+00
    12.    VP1= -.166667+00    VP2=  .235702+00    VP3= -.235702+00

ON CONSTRAINT DISK NO.1
             RMEAN                  AREA
  1      .100000+01            .000000
  2      .100000+01            .000000

ON CONSTRAINT DISK NO.2
             RMEAN                  AREA
  1      .500000+00            .785398+00
  2      .150000+01            .235619+01
```

The second part of the output printout (pages 25 and 26) is the properties of the plume. GAMMA is the half-cone angle of the radial division in the plume. The number on the left column is the index number for the axial regions of the plume. The plume properties are printed two times in IPLUME = 1 and 2 cases. The first time each individual property is grouped by axial and radial region. In the second printout all properties for a given region are listed. The first printout is done in INPUT subroutine, the second is done in OUTPUT subroutine. In IPLUME = 3 case, only the first printout is given, while in IPLUME = 4 case, both printouts are given in the first output and only the second printout is given at the subsequent output.

Output Example 2

AVERAGE NUMBER-DENSITY OF PLUME = .141239+09 PARTS/FT3 OR .498
TOTAL PLUME VOLUME = .11012+03 REX3 NOZZLE EXIT RADIUS = 18
TOTAL RADIANT HEAT RATE = .4096+13 WATTS

OUTPUT FOR PROPERTY 1

1	.93831+04	.10066+05	.10734+05	.12418+05	.11519+05
2	.93831+04	.10065+05	.10733+05	.12418+05	.11523+05
3	.93825+04	.10065+05	.10733+05	.12418+05	.11523+05
4	.93819+04	.10065+05	.10733+05	.12229+05	.10735+05
5	.89835+04	.96881+04	.10740+05	.11757+05	.84811+04
6	.82957+04	.93117+04	.10483+05	.11202+05	.65589+04
7	.80640+04	.90870+04	.10423+05	.10851+05	.64919+04
8	.80629+04	.93178+04	.10530+05	.10481+05	.61854+04
9	.81841+04	.92017+04	.10221+05	.75663+04	.27023+04
10	.12180+05	.11968+05	.86677+04	.29034+04	.85841+01
11	.16538+05	.14869+05	.91193+04	.24599+04	.35314+00
12	.14790+05	.13372+05	.10208+05	.39722+04	.35314+00
13	.12236+05	.11512+05	.94719+04	.38511+04	.35314+00
14	.10485+05	.95910+04	.61219+04	.17682+04	.35314+00

OUTPUT FOR PROPERTY 2

1	.22650+04	.22663+04	.22685+04	.22680+04	.20888+04
2	.22648+04	.22663+04	.22687+04	.22680+04	.20888+04
3	.22648+04	.22663+04	.22687+04	.22680+04	.20888+04
4	.22648+04	.22663+04	.22687+04	.22683+04	.20891+04
5	.22639+04	.22659+04	.22677+04	.22676+04	.21714+04
6	.22648+04	.22661+04	.22667+04	.22667+04	.21824+04
7	.22312+04	.22416+04	.22490+04	.22443+04	.22880+04
8	.21762+04	.22070+04	.22342+04	.22080+04	.20513+04
9	.21698+04	.22029+04	.22151+04	.21952+04	.21403+04
10	.22596+04	.22667+04	.22327+04	.20995+04	.19978+04
11	.23372+04	.23229+04	.22419+04	.20899+04	.18549+04
12	.23307+04	.23147+04	.22225+04	.20674+04	.18440+04
13	.23170+04	.23044+04	.22431+04	.19732+04	.16667+04
14	.22923+04	.22686+04	.22388+04	.20456+04	.16641+04

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

OUTPUT FOR PROPERTY 3

1	.59741+01	.59568+01	.59450+01	.59595+01	.44964+01
2	.59741+01	.59571+01	.59436+01	.59595+01	.44940+01
3	.59741+01	.56570+01	.56916+01	.59595+01	.44940+01
4	.59741+01	.56690+01	.57016+01	.59595+01	.41424+01
5	.59741+01	.59657+01	.59508+01	.59434+01	.41165+01
6	.59741+01	.59741+01	.59578+01	.59277+01	.44171+01
7	.59331+01	.59587+01	.59588+01	.59031+01	.43878+01
8	.58728+01	.59436+01	.59516+01	.58715+01	.43589+01
9	.57637+01	.57973+01	.58242+01	.54749+01	.36583+01
10	.55666+01	.56197+01	.56126+01	.39272+01	.16230+01
11	.55827+01	.56284+01	.56574+01	.28731+01	.30480+00
12	.57432+01	.57336+01	.57630+01	.29394+01	.30480+00

TP - (DEGREES KELVIN)
TG - (DEGREES KELVIN)
N - (PARTS/CM3)
TAUP - (-)
TAUG - (-)
TAU - (-)
RP - (MICRONS)
A/E - (-)
MF - (-)
HZ - (EXIT RADII)

GAMMA =	1.20	2.40	3.60	4.60	6.00
1 TP =	.22650+04	.22663+04	.22685+04	.22680+04	.20888+04
1 TG =	.20802+04	.21000+04	.21263+04	.21662+04	.19655+04
1 N =	.93831+04	.10066+05	.10734+05	.12418+05	.11519+05
1 TAUP=	.46529+01	.49777+01	.53463+01	.60660+01	.40444+01
1 TAUG=	.87378+00	.94974+00	.10691+01	.11415+01	.11801+01
1 TAU=	.55267+01	.59274+01	.64154+01	.72074+01	.52245+01
1 RP =	.59711+01	.59568+01	.59450+01	.59595+01	.44964+01
1 A/E=	.19093+00	.19310+00	.19873+00	.19188+00	.22948+00
1RADFK=	.40142-03	.13468-02	.26018-02	.34318-02	.35572-02
1 HZ =	.40000+00				
2 TP =	.22646+04	.22663+04	.22687+04	.22680+04	.20888+04
2 TG =	.20981+04	.21068+04	.21209+04	.21911+04	.20800+04
2 N =	.93831+04	.10065+05	.10733+05	.12418+05	.11523+05
2 TAUP=	.46285+01	.49650+01	.53427+01	.60660+01	.40399+01
2 TAUG=	.85757+00	.91596+00	.10114+01	.10226+01	.94589+00
2 TAU=	.54861+01	.58810+01	.63541+01	.70885+01	.49858+01
2 RP =	.59741+01	.59571+01	.59436+01	.59595+01	.44940+01
2 A/E=	.18895+00	.18884+00	.19148+00	.17833+00	.19348+00
2RADFK=	.43913-03	.14333-02	.26759-02	.35358-02	.38755-02
2 HZ =	.80000+00				
3 TP =	.22648+04	.22663+04	.22687+04	.22680+04	.20888+04
3 TG =	.21297+04	.21193+04	.20992+04	.21933+04	.22503+04
3 N =	.93825+04	.10065+05	.10733+05	.12418+05	.11523+05
3 TAUP=	.46289+01	.53774+01	.56501+01	.60660+01	.40399+01
3 TAUG=	.83430+00	.86280+00	.90098+00	.67324+00	.72935+00
3 TAU=	.54632+01	.62402+01	.65511+01	.67392+01	.47692+01
3 RP =	.59741+01	.58570+01	.56916+01	.59595+01	.44940+01
3 A/E=	.18549+00	.17408+00	.16951+00	.13574+00	.15686+00
3RADFK=	.48654-03	.15570-02	.25718-02	.28019-02	.44194-02
3 HZ =	.12000+01				
4 TP =	.22648+04	.22663+04	.22687+04	.22683+04	.20891+04
4 TG =	.21257+04	.20973+04	.20667+04	.21530+04	.22552+04
4 N =	.93819+04	.10065+05	.10733+05	.12229+05	.10705+05

In the case of heating rate calculation with IPLUME = 1, the number of events of emission, re-emission and scattering of each region are tabulated. If the anisotropic scattering option is used, the coefficients of the scattering distribution are printed on top of the table, otherwise, the words "isotropic scattering" are printed. A part of this output is shown in output example 3 (page 28).

Output Example 3

ANISOTROPIC SCATTERING

A= 4.0000	B= 15.000	C= 80.000	D=160.000	E=175.000
F= .2000	G= 4.000	H= 15.000	I= 4.200	J=160.000

DISTRIBUTION OF SCATTERINGS AND EMISSIONS THROUGHOUT PLUME

GAMMA =	1.20	2.40	3.60	4.60	6.00
1 NEMIT =	12	37	76	111	93
1 REEMITS =	23	50	110	125	121
1 ISCAT =	76	218	413	567	375
2 NEMIT =	19	42	83	100	119
2 REEMITS =	29	106	150	151	157
2 ISCAT =	115	373	746	726	641
3 NEMIT =	8	41	82	79	110
3 REEMITS =	34	124	162	139	140
3 ISCAT =	173	519	805	831	737
4 NEMIT =	36	120	199	192	400
4 REEMITS =	85	316	441	344	486
4 ISCAT =	433	1556	2352	2449	2045
5 NEMIT =	12	44	68	104	246
5 REEMITS =	57	135	238	280	318
5 ISCAT =	217	672	1252	1366	1191
6 NEMIT =	19	46	99	165	305
6 REEMITS =	56	196	303	373	487
6 ISCAT =	274	1028	1557	1691	1721
7 NEMIT =	15	51	101	173	269
7 REEMITS =	48	155	328	354	452
7 ISCAT =	273	828	1447	1614	1422
8 NEMIT =	12	58	139	159	279
8 REEMITS =	55	235	450	422	539
8 ISCAT =	304	955	1772	1856	1500
9 NEMIT =	43	130	298	329	470
9 REEMITS =	151	483	902	689	1044
9 ISCAT =	546	1881	3393	2502	1201
10 NEMIT =	68	196	334	314	433
10 REEMITS =	260	779	980	705	1016
10 ISCAT =	926	2794	3678	1435	3
11 NEMIT =	415	1075	1034	994	829

The last part of the output is the tabulation of heating flux on each subarea of target surfaces. The area numbers are consistent with the area view vector numbers in the first part of the output. The heating rates are given in both W/cm^2 and Btu/ft^2-sec in two adjacent columns. The average heating rates of groups of NN1 subareas are also given, as shown in output example 4 (page 30).

Output example 5 (page 31) shows the output for view factor calculations. The view factor column lists the computed view factors from the emitting surface to the subarea. The corresponding reciprocal view factor is given on the same line. The view factor of the emitting surface to the entire target surface and its reciprocal are given below the table.

Output Example 4

TOTAL NUMBER OF EMISSION/ABSORPTION = 30000 177375 REX = 185.0 RADIAT = 4.0964+09 WAT

TARGET NO. 1
ON TARGET MAIN SIDE SURFACE

AREA NUMBER	NUMBER OF HITS	AREA NUMBER	HEAT TRANSFER W/CM2	HEAT TRANSFER BTU/SEC-FT2	AREA NUMBER	HIT AREA
1	35.0	1	1.4141+01	1.2456+01	1	1.428571
2	42.0	2	1.6969+01	1.4947+01	2	1.428571
3	67.0	3	2.7069+01	2.3843+01	3	1.428571
4	53.0	4	2.1413+01	1.8861+01	4	1.428571
5	57.0	5	2.3029+01	2.0285+01	5	1.428571
6	58.0	6	2.3433+01	2.0641+01	6	1.428571
7	70.0	7	2.8282+01	2.4911+01	7	1.428571
8	75.0	8	3.0302+01	2.6690+01	8	1.428571
9	60.0	9	2.4241+01	2.1352+01	9	1.428571
10	45.0	10	1.8181+01	1.6014+01	10	1.428571
11	42.0	11	1.6969+01	1.4947+01	11	1.428571
12	39.0	12	1.5757+01	1.3879+01	12	1.428571
	NNI AVERAGE		2.1649+01	1.9069+01		
13	41.0	13	1.6565+01	1.4591+01	13	1.428571
14	58.0	14	2.3433+01	2.0641+01	14	1.428571
15	54.0	15	2.1817+01	1.9217+01	15	1.428571
16	69.0	16	2.7877+01	2.4555+01	16	1.428571
17	64.0	17	2.5857+01	2.2776+01	17	1.428571
18	86.0	18	3.4746+01	3.0605+01	18	1.428571
19	53.0	19	2.1413+01	1.8861+01	19	1.428571
20	56.0	20	2.2625+01	1.9929+01	20	1.428571
21	45.0	21	1.8181+01	1.6014+01	21	1.428571
22	50.0	22	2.0201+01	1.7794+01	22	1.428571
23	52.0	23	2.1009+01	1.8505+01	23	1.428571
24	32.0	24	1.2929+01	1.1388+01	24	1.428571
	NNI AVERAGE		2.2221+01	1.9573+01		
25	25.0	25	1.0101+01	8.8968+00	25	1.428571
26	40.0	26	1.6161+01	1.4235+01	26	1.428571
27	45.0	27	1.8181+01	1.6014+01	27	1.428571
28	60.0	28	2.4241+01	2.1352+01	28	1.428571
29	41.0	29	1.6565+01	1.4591+01	29	1.428571
30	59.0	30	2.3837+01	2.0996+01	30	1.428571
31	53.0	31	2.1413+01	1.8861+01	31	1.428571
32	37.0	32	1.4949+01	1.3167+01	32	1.428571
33	57.0	33	2.3029+01	2.0285+01	33	1.428571

Output Example 5

VIEW FACTOR OUTPUT

TOTAL SAMPLE = 5000 NSTART = 664321 MHITP = 0 MMISS = 4696 MHITG

EAREA = 6.283190+00

TARGET NO. 1

TARGET MAIN SURFACE, L = 1

AREA NUMBER	NO. OF HITS	AREA NUMBER	VIEW FACTOR	HITAREA	AREA NUMBER	VIEW FACTOR BY RECIPROCITY
1	9.	1	1.8000-03	2.1817+00	1	6.2500-04
2	7.	2	1.4000-03	2.1817+00	2	4.8611-04
3	9.	3	1.8000-03	2.1817+00	3	6.2500-04
4	4.	4	8.0000-04	2.1817+00	4	2.7778-04
5	19.	5	3.8000-03	6.5450+00	5	3.9583-03
6	20.	6	4.0000-03	6.5450+00	6	4.1667-03
7	19.	7	3.8000-03	6.5450+00	7	3.9583-03
8	20.	8	4.0000-03	6.5450+00	8	4.1667-03
9	24.	9	4.8000-03	1.0908+01	9	8.3333-03
10	24.	10	4.8000-03	1.0908+01	10	8.3333-03
11	25.	11	5.0000-03	1.0908+01	11	8.6806-03
12	27.	12	5.4000-03	1.0908+01	12	9.3750-03

TOTAL HIT = 207. VIEW FACTOR = 4.1400-02 RECIPR VF = 3.3120-03

4. PROGRAM CHARACTERISTICS

The following characteristics of executing the SRB thermal radiation program are discussed.

- The code is stored on a 7-track tape which can be read into a Univac 1108 Exec 8 computer by the following control cards.

```
@ RUN
@ ASG,T   TAPE1,T,tape number
@ REWIND  TAPE1
@ COPY,G  TAPE1,TPF$
@ FREE  TAPE 1
```

  When either IPLUME = 3 or 4, an additional tape needs to be assigned.

```
@ ASG,T  10,T,SAVE05       (for IPLUME = 3)
@ ASG,T  10,T, (tape number)     (for IPLUME = 4)
```

  The source program and the relocatable elements takes 33 blocks on tape. If the absolute elements are included, the length is extended to 61 blocks. There are two absolute elements, DP and SP, corresponding to dual plume program and single plume program, respectively.

- The program has to be mapped before execution. The control cards for mapping are as follows:

```
@ MAP,IS  DP,DP
   IN MAIN
   LIB  SYS$*MSFC$.
@ XQT  DP
```

  The core storage taken up by the dual plume program is 50 K. The core requirement for the single plume program is about the same (49 K).

- The core storage requirement for the program varies with the array assignments, which can be done by changing the parameters in the PDP element. The values used in the current version of the program on tape are as follows:

  ISEG = 18,   JSEG = 10,   PPT = 10,   NTMAX = 10,
  NRING = 12,   NSIDEA = 250

where

ISEG = number of regions in the plume in the axial direction

JSEG = number of regions in the plume in the radial direction

PPT = number of plume properties

NTMAX = maximum number of target surfaces (NTARGT $\leq$ NTMAX)

NRING = maximum number of divisions in the radial direction on the constraint disks (NRING $\geq$ MAX(NRING1, NRING2))

NSIDEA = maximum number of view point areas on the target surface (NSIDEA $\geq$ NN1*NN2).

Whenever the dimensions are changed, the program elements involving the dimensions need to be recompiled before execution. To recompile the affected program elements, one needs simply to include the following card in the run stream.

@ ADD, P •FORCARDS

- To obtain a listing of the entire program, the control card shown below is used after the source program file has been copied into core. The printout takes 90 pages.

  @ ADD, P •PRTCARDS

- To obtain a punch card deck of the entire program, the control card shown below is used after the source program file has been copied into core. The punch card output is approximately 3700 cards.

  @ ADD, P •PUNCARDS

- The run time of the program depends very much on the case at hand. The following examples serve to indicate the estimate of a run.

  a. When IPLUME = 3 option is used to generate the data tape for a plume, the run time depends on the attenuation and absorption of the plume. To complete a data tape with 100,000 sample trajectories requires 65 minutes for sea level plume with afterburning and 35 minutes for a 72,000 ft plume.

  b. It takes about 10 seconds to read 10,000 sample points from data tape.

  c. It takes 25 minutes to run 25,000 samples in the dual plume calculation (IPLUME = 1) with three target surfaces. The same run takes about 2 minutes if the data were read from a tape (IPLUME = 4).

d. It takes about 4 minutes to compute 10,000 samples in a view factor calculation involving two target surfaces. A similar run with 40,000 samples requires 16 minutes of computer time.

These runs were executed on the Univac 1108 computer at NASA-MSFC.

5. PROGRAM LISTING

The entire source program is listed on pages 36 through 117. The program elements are listed in alphabetical order except the PDP DIMENS, which defines the dimension parameters and the COMMON blocks and is listed first.

SRB PLUME THERMAL RADIATION PROGRAM LISTING

```
                                                                DATE 030576
@ELT,L DIMENS
ELT007 RLIB70 03/05-11:42:51-(0,)
          000   DIM  PROC
          000          PARAMETER ISEG=18, JSEG=10, PPT=10
          000          PARAMETER IJSEG=ISEG*JSEG, IPT=ISEG+1, JPT=JSEG+1, IJPT=IPT*JPT
          000          COMMON/PROPTY/PROP(PPT,IPT,JPT)
          000          COMMON/RESULT/ ID,IDISK,IDAREA,KRING,KPHI,DISTNS,XX(3)
          000          COMMON/IN1/IX,JX,IRGN,JRGN,NSTART,ISO,SIGBET,REX,GAMMA,SIGMA
          000          COMMON/IN2/ H2(IPT),H2CUBE(IPT),H1H2CB(IPT),Z(IPT)
          000          COMMON/IN3/ GAMMAX(JSEG),TANG(JSEG),TANG2(JSEG),CGAMMA(JSEG),
          000         *            P3TAN2(JSEG),TNG2(JPT)
          000          COMMON/IN4/ IREMIT(IJSEG),NEMIT(IJSEG    ),ISCAT(ISEG,JSEG)
          000          COMMON/IN5/TAU(ISEG,JSEG),TAUP(ISEG,JSEG),TAUG(ISEG,JSEG),
          000         *            XNHU(IJSEG),CHIBTA(IJSEG)
          000          COMMON/CNST1/SA,SB,SC,SD,SE,SF,SG,SH,SI,SJ,C1,C2,C3,C4
          000          COMMON/CNST2/A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,AK,A1K
          000          COMMON/CNST3/S1,S2,S3,S4,S5,S6,S7,S8,S9,C1K,C2K,C3K,C4K
          000          COMMON/CNST4/QA,QB,QC,QD,QE,IEMIT,JEMIT,KEMIT,IEVENT,JEVENT,
          000         *             JNDEX,INDEX,KNDEX
          000          COMMON /CNST5/PI,TWOPI,HALFPI,STFBLZ,RADIAT,H,IPLUME
          000          COMMON/TRIG/SINETA,COSETA,TANETA,COST,THETA,THETA1,PHI1,ETA,ALPHA
          000          COMMON/QP/TTNG(JPT),TTNG2(JPT),V(IJSEG)
          000     END
          000   GEOM PROC
          000          PARAMETER NTMAX=10, NRING=12, NSIDEA=250
          000          COMMON/ONCE/ NTARGT,COEF(NTMAX,10),DATA(NTMAX,16),DISKEQ(NTMAX,5)
          000         *,IBODY(NTMAX),RBOND(NTMAX),PLAREA(NTMAX),NAREA(NTMAX,3),
          000         *VECTOR(NTMAX,NSIDEA,3),VAREA(NTMAX,NSIDEA),RRING(NTMAX,2)
          000         *,CRING(NTMAX,NRING,2),NCRING(NTMAX,2),DPHIC(NTMAX),RBONDU(NTMAX),
          000         * RMEANC(NTMAX,NRING,2),SREF(NTMAX,3,2),TREF(NTMAX,3,2)
          000          COMMON /FINAL/ HIT(NSIDEA,NTMAX,3)
          000     END

END ELT.

JPRT,S ATTEN
FURPUR 0026-03/05-11:42
```

```
LEE-ALBIN202*TPFS.ATTEN
            SUBROUTINE ATTEN(IXPR,RI,RIS,H,JZ,ZESCAP)
            INCLUDE DIM,LIST
            J=1
            CALL RANDOM(U)
            XY = -ALOG(U)
            ATEN = 0.0
            SEVENT = 0.0
            IF( COSETA .GT. 0.0) GO TO 300
            IF(JNDEX.LE.JZ) GO TO 500
            IF(COST .LT. 0.0) GO TO 550
      C ATTENUATION LOOP FOR THETA LT 90 DEG AND ETA GT 90 DEG
        500 IF (HZ(INDEX) .GT. Z(J)) GO TO 400
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX,JNDEX+1)
            SMAX = (Z(J) - Z(J-1))/COSETA
            IF (J.EQ.1) SMAX= (Z(1) - H ) / COSETA
        501 SP = (XY - ATEN) /  TAU(INDEX,JNDEX)
            IF (SP. LE. SMAX) GO TO 600
            IF (JNDEX .EQ. JRGN) GO TO 1000
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            JNDEX = JNDEX +1
            SEVENT = SEVENT + SMAX
            J = J+1
            GO TO 500
        400 SMAX = (HZ(INDEX) - Z(J-1))/COSETA
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX-1,JNDEX)
            IF (J.EQ.1) SMAX = (HZ(INDEX)-H )/COSETA
        401 SP = (XY-ATEN)/ TAU(INDEX,JNDEX)
            IF (SP .LE. SMAX) GO TO 600
            IF( INDEX .EQ. 1) GO TO 1000
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX - 1
            SEVENT = SEVENT + SMAX
        402 IF(HZ(INDEX).GT.Z(J)) GO TO 403
            SMAX = (Z(J) - HZ(INDEX+1))/COSETA
            GO TO 501
        403 SMAX = (HZ(INDEX) - HZ(INDEX +1))/ COSETA
            SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
            IF (SP.LE. SMAX) GO TO 600
            IF (INDEX .EQ.1) GO TO 1000
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX-1
            SEVENT = SEVENT + SMAX
            GO TO 402
      C ATTENUATION LOOP FOR THETA GT 90 DEG AND ETA GT 90 DEG
        550 IF (HZ( INDEX ) .GT. Z(J)) GO TO 450
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX,JNDEX-1)
            SMAX = (Z(J) -Z(J-1) )/COSETA
            IF(J.EQ.1) SMAX = (Z(1)-H )/COSETA
        551 SP = (XY-ATEN)/ TAU(INDEX,JNDEX)
            IF(SP.LE.SMAX) GO TO 600
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            JNDEX = JNDEX - 1
            SEVENT = SEVENT + SMAX
            J = J+1
      C IF JNDEX=JZ BUNDLE WILL NOT INTERSECT CONE(JNDEX-1)   USE THETA LT 90 DEG LOOP
```

```
            IF(JNDEX.LE.JZ) GO TO 500
            GO TO 550
        450 SMAX = (HZ(INDEX) -Z(J-1))/COSETA
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX-1,JNDEX)
            IF (J.EQ.1) SMAX = (HZ(INDEX) - H )/COSETA
            SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
            IF(SP .LE. SMAX) GO TO 600
            IF(INDEX .EQ. 1) GO TO 1000
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX - 1
            SEVENT = SEVENT + SMAX
        451 IF(HZ(INDEX) .GT.Z(J)) GO TO 452
            SMAX = (Z(J) - HZ(INDEX +1)) / COSETA
            GO TO 551
        452 SMAX = (HZ(INDEX) - HZ(INDEX +1))/COSETA
            SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
            IF (SP .LE. SMAX) GO TO 600
            IF (INDEX .EQ. 1) GO TO 1000
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX - 1
            SEVENT=SEVENT+SMAX
            GO TO 451
        300 IF(JNDEX.LE.JZ) GO TO 301
            IF(COST.LT.0.0)  GO TO 750
      C ATTENUATION LOOP FOR THETALT 90 DEG AND ETA LT 90 DEG
        301 IF(HZ(INDEX+1).LT.Z(J)) GO TO 800
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX,JNDEX+1)
            SMAX = (Z(J) - Z(J-1)) / COSETA
            IF( J.EQ. 1) SMAX = (Z(1) - H ) / COSETA
        302 SP = (XY - ATEN) /  TAU(INDEX,JNDEX)
            IF (SP .LE. SMAX) GO TO 600
            IF (JNDEX .EQ. JRGN) GO TO 1001
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            JNDEX = JNDEX+1
            SEVENT = SEVENT +SMAX
            J = J+1
            GO TO 301
      C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX+1,JNDEX)
        800 SMAX = (HZ(INDEX+1) - Z(J-1) ) /COSETA
            IF( J.EQ.1) SMAX = (HZ(INDEX+1) - H ) / COSETA
            SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
            IF(SP.LE.SMAX) GO TO 600
            IF (INDEX .EQ. IRGN) GO TO 1001
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX+1
            SEVENT = SEVENT + SMAX
        801 IF (HZ(INDEX+1) .LT. Z(J)) GO TO 802
            SMAX = (Z(J) - HZ(INDEX)) / COSETA
            GO TO 302
        802 SMAX = (HZ(INDEX +1) - HZ(INDEX)) / COSETA
            SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
            IF (SP.LE.SMAX) GO TO 600
            IF (INDEX.EQ. IRGN) GO TO 1001
            ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
            INDEX = INDEX +1
            SEVENT = SEVENT + SMAX
```

```
      GO TO 801
C ATTENUATION LOOP FOR THETA GT 90 DEG AND ETA LT 90 DEG
  750 IF(HZ(INDEX +1) .LT. Z(J)) GO TO 650
C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX,JNDEX-1)
      SMAX = (Z(J) - Z(J-1) ) / COSETA
      IF (J .EQ. 1) SMAX = (Z(1) - H ) / COSETA
  751 SP = (XY - ATEN) /  TAU(INDEX,JNDEX)
      IF(SP .LE.SMAX) GO TO 600
      ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
      JNDEX = JNDEX - 1
      SEVENT = SEVENT + SMAX
      J = J+1
C IF JNDEX=JZ BUNDLE WILL NOT INTERSECT CONE(JNDEX-1)   USE THETA LT 90 DEG LOOP
      IF(JNDEX.LE.JZ) GO TO 301
      GO TO 750
C BUNDLE FROM REGION(INDEX,JNDEX) INTERSECTS REGION(INDEX+1,JNDEX)
  650 SMAX = (HZ(INDEX +1) - Z(J-1)) / COSETA
      IF (J .EQ. 1) SMAX = (HZ(INDEX +1) - H ) / COSETA
      SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
      IF (SP .LE. SMAX) GO TO 600
      IF (INDEX .EQ. IRGN) GO TO 1001
      ATEN = ATEN +  TAU(INDEX,JNDEX)*SMAX
      INDEX = INDEX+1
      SEVENT = SEVENT + SMAX
  651 IF (HZ(INDEX +1) .LT. Z(J)) GO TO 652
      SMAX = (Z(J) - HZ(INDEX)) / COSETA
      GO TO 751
  652 SMAX = (HZ(INDEX+1) - HZ( INDEX) ) / COSETA
      SP = (XY-ATEN) /  TAU(INDEX,JNDEX)
      IF (SP.LE. SMAX) GO TO 600
      IF (INDEX .EQ. IRGN) GO TO 1001
      ATEN = ATEN +  TAU (INDEX,JNDEX) * SMAX
      INDEX = INDEX+1
      SEVENT = SEVENT + SMAX
      GO TO 651
  600 CONTINUE
C CALCULATE COORDINATES OF SCATER EVENT
      IEVENT = INDEX
      JEVENT = JNDEX
      SEVENT = SEVENT +SP
      SESNET = SEVENT*SINETA
      H=H+COSETA*SEVENT
      SS=SESNET**2
      R2S=R1S+SS+2.*SESNET*R1*COS(ALPHA)
      R2=SQRT(R2S)
      ARGUMT=(R1S+R2S-SS)/(2.*R1*R2)
      IF (ARGUMT.LT.-1.)                       ARGUMT=-.99999
      IF (ARGUMT.GT.1.0)                       ARGUMT=0.99999
  201 BETA=ACOS(ARGUMT)
      PHI1=PHI1+BETA
      IF (ALPHA.GT.PI)                         PHI1=PHI1-2.*BETA
      IF (PHI1.GT.TWOPI)PHI1=PHI1-TWOPI
      IF(PHI1.LT.0.0)PHI1=PHI1+TWOPI
  202 R1S=R2S
      R1=R2
      REDGE=H*TANG(JNDEX)
```

```
      IF(REDGE.GT.R1) GO TO 1112
      JNDEX=JNDEX+1
      IF(JNDEX.LE.JRGN) GO TO 1112
      IXPR=1
      IF(COSETA.GT.0.0) IXPR=2
      ZESCAP=Z(J+1)
      SIGBET=SEVENT
      RETURN
 1112 KNDEX=JRGN*(INDEX-1)+JNDEX
      CALL RANDOM(U)
      IF(U.LT.CHIBTA(KNDEX)) GO TO 200
      ISCAT(    INDEX,JNDEX)=ISCAT(    INDEX,JNDEX) + 1
      IXPR=3
      RETURN
  200 IREMIT(KNDEX)=IREMIT(KNDEX)+1
      IXPR=3
      RETURN
 1000 IXPR=1
      ZESCAP=Z(J)
      SIGBET=SEVENT+SMAX
      RETURN
 1001 IXPR=2
      ZESCAP=Z(J)
      SIGBET=SEVENT+SMAX
      RETURN
      END
```

```
LEE-ALBIN202*TPFS.CHOSE
      SUBROUTINE CHOSE (P1,P2,ROOTS,SMIN,IOUT)
      DIMENSION P1(3),P2(3),ROOTS(4,2),SMIN(4),INSIDE(2)
      RANGE(X1,Y1,Z1,X2,Y2,Z2)=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
      IOUT=0
      EPS=1.0E-4
      AL=RANGE(P1(1),P1(2),P1(3),P2(1),P2(2),P2(3))
      DO 10 II=1,2
      TX=ROOTS(1,II)
      TY=ROOTS(2,II)
      TZ=ROOTS(3,II)
      AL1=ABS((TX-P1(1))*(P2(1)-P1(1))+(TY-P1(2))*(P2(2)-P1(2))
     *     +(TZ-P1(3))*(P2(3)-P1(3)))/AL
      AL2=ABS((TX-P2(1))*(P2(1)-P1(1))+(TY-P2(2))*(P2(2)-P1(2))
     *     +(TZ-P2(3))*(P2(3)-P1(3)))/AL
      IF (ABS((AL-(AL1+AL2))/AL).GT.EPS) GO TO 20
    1 INSIDE(II)=1
      GO TO 10
   20 INSIDE(II)=0
   10 CONTINUE
      IF (INSIDE(1).EQ.0.AND.INSIDE(2).EQ.0) GO TO 23
      IF (INSIDE(1).EQ.0) GO TO 21
      IF (INSIDE(2).EQ.0) GO TO 22
      IF (ROOTS(4,1).GT.ROOTS(4,2)) GO TO 21
   22 SMIN(1)=ROOTS(1,1)
      SMIN(2)=ROOTS(2,1)
      SMIN(3)=ROOTS(3,1)
      SMIN(4)=ROOTS(4,1)
      GO TO 100
   21 SMIN(1)=ROOTS(1,2)
      SMIN(2)=ROOTS(2,2)
      SMIN(3)=ROOTS(3,2)
      SMIN(4)=ROOTS(4,2)
      GO TO 100
   23 IOUT=1
  100 RETURN
      END

@PRT,S DIFVDC
```

```
LEC-ALBIN202*TPF$.DIFVOC
      SUBROUTINE DIFVOC(SU,PHRANG,U)
C
C *** MODIFIED DIFVOC FOR RADIATIVE HEAT TRANSFER PROGRAM TESTING PURPOSES.
C
      DIMENSION SU(3),U(3)
      PI=3.14159
      CALL RANDOM (RN1)
      CALL RANDOM (RN2)
      TH=PI*2.*RN2
      HALFPI=PI/2.0
      IF (PHRANG.GT.3.00)                    GO TO 30
      PH=ASIN(SQRT(RN1))
      DIFF=ABS(PHRANG-HALFPI)
      IF (DIFF.LT.0.1)                       GO TO 20
      PH=PH*PHRANG/HALFPI
   20 CONTINUE
      U1P=COS(PH)
      U2P=SIN(PH)*COS(TH)
      U3P=SIN(PH)*SIN(TH)
      DEL=SU(2)*SU(2)+SU(3)*SU(3)
      IF (DEL.LT. 1.E-10)                    SU(2)=0.0001
      PHS=ACOS(SU(1))
      THS=HALFPI+ATAN2(SU(3),SU(2))
      U(1) = U1P*COS(PHS)                    +U3P*SIN(PHS)
      U(2) = U1P*SIN(PHS)*SIN(THS)+U2P*COS(THS)-U3P*COS(PHS)*SIN(THS)
      U(3) =-U1P*SIN(PHS)*COS(THS)+U2P*SIN(THS)+U3P*COS(PHS)*COS(THS)
      GO TO 50
   30 PH=ACOS(1.-RN1-RN1)
      U 1)=COS(PH)
      U(2)=SIN(PH)*COS(TH)
      U(3)=SIN(PH)*SIN(TH)
   50 RETURN
      END

*PRT,S DISK
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
LEL-ALBIN202*TPFS.DISK
      SUBROUTINE DISK (IFOR,NDATA,P,A,XT,IHIT)
C     IFOR=0, READ INPUT DATA
C     IFOR=1, CHECK FOR HIT
C     IHIT EQ 0, NO HIT ON THE DISK
C     IHIT GE 0, SCORE A HIT, EQUAL TO NSIDEA NUMBER
      INCLUDE GEOM,LIST
      DIMENSION A(3),P(3),XT(4)
      DETMNT(X11,X12,X13,X21,X22,X23,X31,X32,X33) =
     1                         X11*X22*X33 + X21*X32*X13 + X31*X12*X23
     2                       - X11*X23*X32 - X22*X13*X31 - X33*X12*X21
      IHIT=0
      XT(4)=2.E30
      PI=3.14159
      TWOPI=6.28318
      NA=INT(DATA(NDATA,1))
      NR=INT(DATA(NDATA,2))
      ROUT=DATA(NDATA,4)
      X1=DATA(NDATA,5)
      Y1=DATA(NDATA,6)
      Z1=DATA(NDATA,7)
      X2=DATA(NDATA,8)
      Y2=DATA(NDATA,9)
      Z2=DATA(NDATA,10)
      A1=DATA(NDATA,11)
      A2=DATA(NDATA,12)
      A3=DATA(NDATA,13)
      RIN=RRING(NDATA,1)
      DR=(ROUT-RIN)/FLOAT(NR)
      DT=TWOPI/FLOAT(NA)
      RMAG=SQRT((X2-X1)**2 + (Y2-Y1)**2 + (Z2-Z1)**2)
      B1=(X2-X1)/RMAG
      B2=(Y2-Y1)/RMAG
      B3=(Z2-Z1)/RMAG
      AB1=A2*B3-A3*B2
      AB2=A3*B1-A1*B3
      AB3=A1*B2-A2*B1
      IF (IFOR-1)                                  100,200,200
  100 CONTINUE
      WRITE (6,190)
      DISCR=DETMNT(B1,B2,B3,A1,A2,A3,AB1,AB2,AB3)
      RMAG=RIN-DR/2.0
      NSIDEA=0
      NRR=0
  110 RMAG=RMAG+DR
      IF (RMAG.GT.ROUT)                            GO TO 150
      AREA=RMAG*DR*DT
      TH=-DT/2.0
      NSIDEX=NRR*NA
      NRR=NRR+1
  120 TH=TH+DT
      IF (TH.GT.TWOPI)                             GO TO 110
      NSIDEX=NSIDEX+1
      SINTH=SIN(TH)
      COSTH=COS(TH)
      C1=DETMNT(COSTH,B2,B3,0.,A2,A3,SINTH,AB2,AB3)/DISCR
```

```
      C2=DETMNT(B1,COSTH,B3,A1,0.,A3,AB1,SINTH,AB3)/DISCR
      C3=DETMNT(B1,B2,COSTH,A1,A2,0.,AB1,AB2,SINTH)/DISCR
      CMAG=SQRT(C1*C1+C2*C2+C3*C3)
      VECTOR(NDATA,NSIDEX,1)=X1+RMAG*C1/CMAG
      VECTOR(NDATA,NSIDEX,2)=Y1+RMAG*C2/CMAG
      VECTOR(NDATA,NSIDEX,3)=Z1+RMAG*C3/CMAG
      VAREA(NDATA,NSIDEX)=AREA
      RNSIDE=FLOAT(NSIDEX)
      WRITE (6,191) RNSIDE,(VECTOR(NDATA,NSIDEX,I),I=1,3),
     *                 VAREA(NDATA,NSIDEX)
      GO TO 120
  150 CONTINUE
      GO TO 500
  190 FORMAT (///39H VIEW POINT VECTORS OF AN ANNULAR DISK  /)
  191 FORMAT (1X,F7.0,4X,4HVP1=,E12.6,3X,4HVP2=,E12.6,3X,4HVP3=,E12.6,
     *        8X,SHAREA=,E12.6 )
  200 CONTINUE
      DENOM=A1*A(1)+A2*A(2)+A3*A(3)
      IF (ABS(DENOM).LT.1.E-20)                        GO TO 500
      DN = - (A1*(P(1)-X1)+A2*(P(2)-Y1)+A3*(P(3)-Z1)) / DENOM
      IF (DN.LT.0.)                                    GO TO 500
      XT(1)=P(1)+A(1)*DN
      XT(2)=P(2)+A(2)*DN
      XT(3)=P(3)+A(3)*DN
      RXT=SQRT((XT(1)-X1)**2+(XT(2)-Y1)**2+(XT(3)-Z1)**2)
      IF (RXT.GT.ROUT .OR. RXT.LT.RIN)                 GO TO 500
      XT(4)=DN
      NRX=INT((RXT-RIN)/DR)
      C1=(XT(1)-X1)/RXT
      C2=(XT(2)-Y1)/RXT
      C3=(XT(3)-Z1)/RXT
      TH=ACOS(C1*B1+C2*B2+C3*B3)
      DET=DETMNT(C1,C2,C3,A1,A2,A3,B1,B2,B3)
      IF (DET.LT.0.)                                   TH=TWOPI-TH
      NTH=INT(TH/DT)+1
      IHIT=NRX*NA + NTH
  500 CONTINUE
      RETURN
      END
```

```
$PRT,S EMITT
```

```
LEE-ALBIN202*TPFS.EMITT
            SUBROUTINE EMITT(NRGN,X,U,H,RIS,RI)
            INCLUDE DIM,LIST
            DO 100 I = 1,NRGN
            X=X+XNHU(    I)
            XD = X/U
            IF(XD.GT.1.) GO TO 200
        100 CONTINUE
        200 IEMIT = ( I - 1 )/JRGN + 1
            JEMIT = I - JRGN * (IEMIT-1)
            INDEX = IEMIT
            JNDEX = JEMIT
            IEVENT = IEMIT
            JEVENT = JEMIT
            KEMIT = 1
            KNDEX = 1
      C FIX EMISSION COORDINATES
            CALL RANDOM(U)
            H = ( HIH2CB(IEMIT)*U + H2CUBE(IEMIT))**.33333
            CALL RANDOM(U)
            RIS = H**2*(U*(TNG2(JEMIT+1)-TNG2(JEMIT))+TNG2(JEMIT))
            RI = SQRT(RIS)
      C SELECT EMISSION DIRECTION
            CALL RANDOM(U)
            THETA=TWOPI*U
            CALL RANDOM(U)
            COSETA = 1. - 2.*U
            SINETA = SQRT( 1. - COSETA**2 )
            TANETA = SINETA / COSETA
            C LL RANDOM(U)
            PHI1=TWOPI*U
            RETURN
            END

@PRT,S ESCAP
```

```
LEE-ALBIN207*TPFS.ESCAP
              SUBROUTINE ESCAP (X,A)
              INCLUDE GEOM,LIST
              INCLUDE DIM,LIST
      C     POSITIVE  X(1)  IS IN THE PLUME EXIT DIRECTION.
      C     SET THE ORIGIN AT THE CENTER OF THE PLUME BASE PLANE.
              DIMENSION X(3),A(3)
              NEMIT(KEMIT) = NEMIT(KEMIT) + 1
              IF (IPLUME.EQ.3)                                GO TO 450
              CALL SORTNG (X,A)
              IF (ID.EQ.0)                                    GO TO 300
              DO 100 ITG=1,NTARGT
              IF (ITG.NE.ID)                                  GO TO 100
              DO 110 L=1,3
              IF (L.NE.(IDISK+1))                             GO TO 110
              NA=NAREA(ID,L)
              DO 120 INA=1,NA
              IF (INA.NE.IDAREA)                              GO TO 120
              HIT(INA,ITG,L)=HIT(INA,ITG,L)+1.
          120 CONTINUE
          110 CONTINUE
          100 CONTINUE
          300 CONTINUE
              M=M+1
              GO TO 500
          450 IF (M.GE.1)                                     GO TO 455
              WRITE (6,451)
          455 STOP
          451 FORMAT (////10X,45H IPLUME=3 IN DUAL PL IS NOT APPLICABLE         )
          500 RETURN
              END

@PRT,S ESCAPE
```

```
LEE-ALBIN202*TPFS.ESCAPE
      SUBROUTINE ESCAPE(H,H1,IXPR,R1,R15,ZESCAP)
      INCLUDE GEOM,LIST
      INCLUDE DIM,LIST
      DIMENSION X(3),A(3)
C   SET THE ORIGIN AT THE CENTER OF THE PLUME BASE PLANE.
C   POSITIVE  X(1)  IS IN THE PLUME EXIT DIRECTION.
      NEMIT(KEMIT) = NEMIT(KEMIT) + 1
      X(1)=H-H1
      A(1)=COSETA
      A(2)=SINETA*COS(THETA)
      A(3)=SINETA*SIN(THETA)
      R=R1
      X(2)=R*COS(PHI1)
      X(3)=R*SIN(PHI1)
      ETA=ACOS(COSETA)
      IF (IPLUME.EQ.3)                                GO TO 450
      CALL SORING (X,A)
      IF (ID.EQ.0) GO TO 201
      DO 100 ITG=1,NTARGT
      IF (ITG.NE.ID) GO TO 100
      DO 400 L=1,3
      IF (L.NE.(IDISK+1)) GO TO 400
      NA=NAREA(ID,L)
      DO 200 INA=1,NA
      IF (INA.NE.IDAREA) GO TO 200
      HIT(INA,ITG,L)=HIT(INA,ITG,L)+1.
  200 CONTINUE
  400 CONTINUE
  100 CONTINUE
      GO TO 300
  201 CONTINUE
  300 CONTINUE
      M=M+1
      GO TO 500
  450 CONTINUE
      IF (M.GE.1)                                     GO TO 455
      NRGN=IRGN*JRGN
      WRITE (10) IRGN,JRGN,RADIAT,REX
      DO 451 K=1,10
      DO 451 I=1,IRGN
  451 WRITE (10) (PROP(K,I,J), J=1,JRGN)
      WRITE (10) (GAMMAX(J), J=1,JRGN)
      K=0
      DO 452 I=1,IRGN
      WRITE (10) (TAUP(I,J), J=1,JRGN)
      WRITE (10) (TAUG(I,J), J=1,JRGN)
      WRITE (10) (TAU (I,J), J=1,JRGN)
      WRITE (10) (CHIBTA(K+J),J=1,JRGN)
      WRITE (10) (XNHU  (K+J),J=1,JRGN)
      HZ(I+1)=HZ(I+1)-H1
      WRITE (10) HZ(I+1)
      HZ(I+1)=HZ(I+1) + H1
  452 K=K+JRGN
  455 CONTINUE
      X(1)=ZESCAP-H1
```

```
      RXIT=ZESCAP*TANG(JRGN)
      X(2)=X(2)+SIGBET*A(2)
      X(3)=X(3)+SIGBET*A(3)
      PHIX=ATAN2(X(3),X(2))
      IF (PHIX.LT.0.)                              PHIX=PHIX+TWOPI
      THETAX=THETA-PHIX
      IF (THETAX.LT.0.)                            THETAX=THETAX+TWOPI
      IRMIT=0
      DO 456 I=1,NRGN
  456 IRMIT=IRMIT+IREMIT(I)
      CALL IOPKT (M,IRMIT,X(1),RXIT,ETA,THETAX,PHIX,1)
      IF (M.GE.IX*JX)                              ENDFILE 10
  500 RETURN
      END
```

```
-PRT,S FRTAPE
```

```
LEE-ALBIN202*TPFS.FRTAPE
      SUBROUTINE FRTAPE (HI,NRGN,NPLM)
C     READ DATA TAPE WITH IOPKT (M IRMIT Z R E T P IO)
C --- KEY=1, COMPUTE THE HEATING RATE ON GEOMETRY ENSEMBLE
C --- KEY=2, SMALL TARGET AROUNG THE PLUME, DELPHI=2PI
C --- KEY=3, SINGLE SMALL TARGET ENTIRELY IN DELPHI,DELPHI IN DEGREES.
C --- KEY=4, SINGLE SMALL TARGET, USE COSINE PROJECTION OF DELPHI
C     TX,TY,TZ, IS CENTER OF SMALL GEOMETRY
C     NPLM=1, SINGLE PLUME, 2, DUAL PLUME, NTRAJ IS NO OF PAIRS OF TRAJ
      INCLUDE GEOM,LIST
      INCLUDE DIM,LIST
      DIMENSION XP(3),PP(3)
      DIMENSION X(3),P(3),TEMHIT(NSIDEA,3),IPLME(12),JPLME(12),KPLME(12)
      DIMENSION IZ(12),LTRU(12)
      PI=3.14159
      TWOPI=6.28318
      KIO=0
   50 READ (5,10) KEY,ITG,NTRAJ,DELPHI,TX,TY,TZ
      IF (NTRAJ.EQ.0)                                GO TO 900
      REWIND 10
      KIO=KIO+1
      READ (10)  IRGN,JRGN,RADIAT,REX
      NRGN=IRGN*JRGN
      DO 60 K=1,10
      DO 60 I=1,IRGN
   60 READ (10) (PROP(K,I,J), J=1,JRGN)
      READ (10) (GAMMAX(J), J=1,JRGN)
      GAMXX=GAMMAX(JRGN)
      TANG(JRGN)=TAN(GAMXX)
      K=0
      DO 61 I=1,IRGN
      READ  (10) (TAUP(I,J), J=1,JRGN)
      READ  (10) (TAUG(I,J), J=1,JRGN)
      READ  (10) (TAU (I,J), J=1,JRGN)
      READ  (10) (CHIBTA(K+J),J=1,JRGN)
      READ  (10) (XNHU  (K+J),J=1,JRGN)
      READ  (10) HZ(I+1)
   61 K=K+JRGN
      WRITE (6,11) KEY,ITG,NTRAJ,DELPHI,TX,TY,TZ
      WRITE (6,12) IRGN,JRGN,RADIAT,REX
      IF (KEY.GT.0)                                  GO TO 75
      WRITE (6,13)
      WRITE (6,14)
      DO 70 K=1,10
      WRITE (6,17) K
      DO 70 I=1,IRGN
   70 WRITE (6,18) I,(PROP(K,I,J), J=1,JRGN)
 1400 FORMAT(/// 40H1            PROPERTIES IN PLUME REGIONS      // )
 1402 FORMAT(  //  26H TP - ( DEGREES KELVIN )      /
     1 26H TG - ( DEGREES KELVIN )           /
     1 26H N - ( PARTS/CM3 )     /
     - 15H TAUP - ( - )          /
     * 15H TAUG - ( - )          /
     2 15H TAU - ( - )           /
     3 20H RP - ( MICRONS )      /
     4 15H A/E - ( - )           /
```

```
     5 16H RADFK = ( = )                          /
     7 30H HZ = ( EXIT RADII )                               /// )
1403 FORMAT ( 10H   GAMMA =, 11E11.2)
1405 FORMAT (/I4,6H   TP =, 11E11.5    )
8002 FORMAT ( I4,6H   TG =, 11E11.5    )
1406 FORMAT ( I4,6H   N  =, 11E11.5    )
1812 FORMAT ( I4,6H TAUP=, 11E11.5     )
1813 FORMAT ( I4,6H TAUG=, 11E11.5     )
1407 FORMAT ( I4,6H  TAU=, 11E11.5     )
1408 FORMAT ( I4,6H   RP =, 11E11.5    )
1409 FORMAT ( I4,6H  A/E=, 11E11.5     )
1410 FORMAT ( I4,6HRADFK=, 11E11.5     )
1412 FORMAT ( I4,6H  HZ =, 11E11.5 //)
  75 CONTINUE
     IF (K10.GT.1)                                          GO TO 78
     WRITE(6,1400)
     WRITE (6,1402)
     WRITE( 6,1403 ) ( GAMMAX(J) , J=1,JRGN)
     K=0
     DO 1404 I= 1,IRGN
     WRITE(6,1405) I,(PROP(2,I,J),J=1,JRGN)
     WRITE (6,8002) I,(PROP(5,I,J),J=1,JRGN)
     WRITE(6,1406) I,(PROP(1,I,J),J=1,JRGN)
     WRITE(6,1812) I, (TAUP(I,J),J=1,JRGN)
     WRITE(6,1813) I, (TAUG(I,J),J=1,JRGN)
     WRITE (6,1407) I, (TAU (I,J), J=1,JRGN)
     WRITE (6,1408) I,(PROP(3,I,J),J=1,JRGN)
     WRITE( 6,1409) I , ( CHIBTA(K+J), J = 1,JRGN )
     WRITE(6,1410) I, (XNHU(  K+J), J=1,JRGN)
     HZ(I+1) = HZ(I+1) - H1
     WRITE(6,1412) I , HZ(I+1)
     HZ(I+1) = HZ(I+1) + H1
1404 K=K+JRGN
  10 FORMAT (3I8,6F8.0)
  11 FORMAT (1H1,21HREAD FROM TAPE  KEY=  ,I2,5X,11HTARGET NO = ,I3,3X,
    1          ,11HSAMPLE NO = ,I7,3X,8HDELPHI = ,F6.2,5H DEG. ,5X,
    2          10HTX,TY,TZ = ,3F7.3  /)
  12 FORMAT (7H IRGN = ,I3,3X,6HJRGN = ,I3,10X,
    1          25HTOTAL RADIANT HEAT RATE = ,1PE10.4,7H WATTS ,
    2          6X,5HREX =,0PF8.2,3H CM  // )
  13 FORMAT (10X,50HSUBSCRIPTION K IN PROP(K,I,J) ARRAY
    1        //10X,50HK=1, PARTICLE NUMBER DENSITY (P/CM**3)
    2          ,5X,50HK=2, PARTICLE TEMPERATURE (DEG KELVIN)
    3          /10X,50HK=3, PARTICLE RADIUS (MICRONS)
    4          ,5X,50HK=4, PROJECTED AREA OF PARTICLE (MICRON**2)       )
  14 FORMAT (10X,50HK=5, GAS TEMPERATURE (DEG KELVIN)
    1          ,5X,50HK=6, STATIC PRESSURE (PSIA)
    2          /10X,50HK=7, MOLE FRACTION FOR CO  (-)
    3          ,5X,50HK=8  MOLE FRACTION FOR CO2 (-)
    4          /10X,50HK=9, MOLE FRACTION FOR H2O-(-)
    5          ,5X,50HK=1, MOLE FRACTION FOR HCL (-)                   )
  17 FORMAT (/// 17H OUTPUT FOR PROP  ,5X,3HI =,I3 /)
  18 FORMAT (I3,1P10E12.4  )
  20 FORMAT (I10,5X,1P5E14.5 )
  78 CONTINUE
     DELPHI=DELPHI/57.29578
```

```
      DELHIT=1.0
      IF (KEY.EQ.4)                                    DELHIT=DELPHI/TWOPI
      IMISS=0
      ISCORE=0
      IF (KEY.GT.0)                                    GO TO 90
      IF (KEY.EQ.0)                                    GO TO 900
      N=0
      DO 81 I=1,12
      JPLME(I)=0
      KPLME(I)=0
      LTRU(I)=0
      IZ(I)=0
   81 IPLME(I)=0
      IOUT=0
      JOUT=0
      KOUT=0
      LOUT=0
      DPH=TWOPI/12.0
      DET=PI/12.0
   82 PH=0.0
      ET=0.0
      N=N+1
      IF (N.GT.NTRAJ)                                  GO TO 88
      I=1
      J=1
   83 CALL IOPKT (M,IRMIT,ZS,R,ETA,THETA,PHI,2)
   85 CONTINUE
      IF(ZS.LT.0.0) IZ(1)=IZ(1)+1
      IF(ZS.GE.0.0.AND.ZS.LT.2.0) IZ(2)=IZ(2)+1
      IF(ZS.GE.2.0.AND.ZS.LT.4.0) IZ(3)=IZ(3)+1
      IF(ZS.GE.4.0.AND.ZS.LT.6.0) IZ(4)=IZ(4)+1
      IF(ZS.GE.6.0.AND.ZS.LT.8.0) IZ(5)=IZ(5)+1
      IF(ZS.GE.8.0.AND.ZS.LT.10.) IZ(6)=IZ(6)+1
      IF(ZS.GE.10..AND.ZS.LT.12.) IZ(7)=IZ(7)+1
      IF(ZS.GE.12..AND.ZS.LT.14.) IZ(8)=IZ(8)+1
      IF(ZS.GE.14..AND.ZS.LT.16.) IZ(9)=IZ(9)+1
      IF(ZS.GE.16..AND.ZS.LT.18.) IZ(10)=IZ(10)+1
      IF(ZS.GE.18..AND.ZS.LT.20.) IZ(11)=IZ(11)+1
      IF(ZS.GE.20.0)IZ(12)=IZ(12)+1
      T1=PH
      T2=ET
      TRUPH=PHI+THETA
      IF (TRUPH.GT.TWOPI)                              TRUPH=TRUPH-TWOPI
      DO 86 I=1,12
      IF (PHI.GT.T1 .AND. PHI.LT.(T1+DPH))             IPLME(I)=IPLME(I)+1
      IF (ETA.GT.T2 .AND. ETA.LT.(T2+DET))             JPLME(I)=JPLME(I)+1
      IF (THETA.GT.T1 .AND. THETA.LT.(T1+DPH))         KPLME(I)=KPLME(I)+1
      IF (TRUPH.GT.T1 .AND. TRUPH.LT.(T1+DPH))         LTRU(I)=LTRU(I)+1
      T1=T1+DPH
   86 T2=T2+DET
      IF (PHI.GT.TWOPI)                                IOUT=IOUT+1
      IF (ETA.GT.PI)                                   JOUT=JOUT+1
      IF (THETA.GT.TWOPI)                              KOUT=KOUT+1
      IF (TRUPH.GT.TWOPI)                              LOUT=LOUT+1
      GO TO 82
   88 N=N-1
```

```
      WRITE (6,89) (IPLME(I),I=1,12),(JPLME(J),J=1,12),(KPLME(I),I=1,12)
     1      ,(LTRU(I),I=1,12),(IZ(I),I=1,12),IOUT,JOUT,KOUT,LOUT,N,NTRAJ
   89 FORMAT (1H1, 21HNO OF EMIT IN DEL PHI   ,1X,12I9 //
     1           1X,21HNO OF EMIT IN DEL ETA   ,1X,12I9 //
     2           1X,21HNO OT EMIT IN THETA     ,1X,12I9 //
     3           1X,21HNO OF EMIT IN TRU PHI   ,1X,12I9//
     4           1X,21HNO OF EMIT IN DEL  Z    ,1X,12I9 ////
     5           1X,27HIOUT JOUT KOUT LOUT N NTRAJ  ,7I9 )
      GO TO 50
C ***
   90 KNT=0
      M=0
      IHITP1=0
      IHITP2=0
      DO 80 I=1,NSIDEA
      DO 80 J=1,NTARGT
      DO 80 K=1,3
   80 HIT(I,J,K)=0.0
   91 KNT=KNT+1
      IF (KNT.GT.NTRAJ)                                GO TO 500
   93 CALL IOPKT (M,IRMIT,ZS,R,ETA,THETA,PHI,2)
      IPASS=0
      IF (ZS.LT.0.)                                    GO TO 98
      IF (KEY.EQ.1 .OR. KEY.EQ.2)                      GO TO 96
      PH2=ATAN2(TZ,TY)
      IF (KEY.EQ.4)                                    PHI=PH2
      IF (KEY.EQ.4)                                    GO TO 96
      PHDIF=ABS(PH2-PHI)
      IF (PHDIF.GT.DELPHI/2.)                          GO TO 91
   96 X(1)=ZS
      X(2)=R*COS(PHI)
      X(3)=R*SIN(PHI)
      PHISAV=PHI
      PHI=PHI+THETA
      P(1)=COS(ETA)
      P(2)=SIN(ETA)*COS(PHI)
      P(3)=SIN(ETA)*SIN(PHI)
      IF (NPLM.EQ.1)                                   GO TO 97
      IPASS=1
      KFPLM=1
   92 DO 94 I=1,3
      XP(I)=X(I)
   94 PP(I)=P(I)
      CALL TRANSF (XP,X,PP,P,KFPLM,0)
      CALL INTRCP (X,P,KFPLM,KOUT,XP,PP)
      IF (KOUT.EQ.0)                                   GO TO 97
      IF (KOUT.EQ.1)                                   IHITP1=IHITP1+1
      IF (KOUT.EQ.2)                                   IHITP2=IHITP2+1
      IF (IPASS.EQ.2)                                  GO TO 91
      GO TO 115
C
   97 CONTINUE
      CALL SORTNG (X,P)
C
      IF (ID.GT.0)                                     GO TO 100
   98 IMISS=IMISS+1
```

```
      IF (IPASS.GT.0)                                 GO TO 115
      IF (NPLM.EQ.1)                                  GO TO 91
      IMISS=IMISS+1
      GO TO 91
  100 CONTINUE
      IF (KEY.GT.1 .AND. ID.NE.ITG)                   GO TO 115
      INA=IDAREA
      IITG=ID
      LDIS=IDISK+1
      HIT(INA,IITG,LDIS)=HIT(INA,IITG,LDIS)+DELHIT
      ISCORE=ISCORE+1
  115 IF (NPLM.EQ.1 .OR. IPASS.EQ.2)                  GO TO 91
      IPASS=2
      KFPLM=2
      PHI=PI-PHISAV
      X(1)=ZS
      X(2)=R*COS(PHI)
      X(3)=R*SIN(PHI)
      PHI=PHI-THETA
      P(1)=COS(ETA)
      P(2)=SIN(ETA)*COS(PHI)
      P(3)=SIN(ETA)*SIN(PHI)
      GO TO 92
  500 CONTINUE
      WRITE (6,510) KEY,ITG,NTRAJ,ISCORE,IHITP1,IHITP2,IMISS,IRMIT
  510 FORMAT (///6H KEY =,I3,5X,5HITG =,I3,5X,7HNTRAJ =,I8,5X,8HISCORE =
     1       ,I7,5X,10HIHITP1/2 =,2I8,5X,7HIMISS =,I7,5X,7HIRMIT =,I7/)
  120 IF (KEY.NE.4)                                   GO TO 600
      DO 121 I=1,NSIDEA
      DO 121 J=1,3
  121 TEMHIT(I,J)=0.0
      PH=0.
  122 PH=PH+DELPHI
      COSPH=COS(PH)
      IF (COSPH.LT.0.)                                GO TO 130
      DO 125 LDIS=1,3
      NA=NAREA(ITG,LDIS)
      DO 125 K=1,NA
      TEMHIT(K,LDIS) = TEMHIT(K,LDIS)+HIT(K,ITG,LDIS)*2.*COSPH
  125 CONTINUE
      GO TO 122
  130 DO 132 I=1,3
      DO 132 J=1,NSIDEA
  132 HIT(J,ID,I)=TEMHIT(J,I)
  600 CONTINUE
C *** OUTPUT THE RESULTS
      IREMIT(1)=IRMIT
      IF (KEY.GT.1)                                   GO TO 610
      CALL OUTPUT (HI,NRGN,0)
      GO TO 900
  610 CALL OUTPUT (HI,NRGN,ITG)
      GO TO 50
  900 RETURN
      END
```

```
LIE-ALBIN202*TPFS.INPUT
      SUBROUTINE INPUT (NRGN,HI,FACTR)
C     FACTR=1 FOR SINGLE PLUME, 2 FOR DUAL PLUME IN COMPUTING RADIA.
      INCLUDE DIM,LIST
  899 FORMAT (7I8,3F8.0)
  885 FORMAT ( 8F10.2 )
  772 FORMAT (35H1AVERAGE NUMBER DENSITY OF PLUME = ,E15.6,1X,
     1        15HPARTS/FT3  OR    ,E15.5,10H PART/CM3   /
     2        21H TOTAL PLUME VOLUME =  ,E15.5,5H REX3,10X,
     3        21H NOZZLE EXIT RADIUS =  ,F8.2,3H CM  /
     4        26H TOTAL RADIANT HEAT RATE =  ,E13.4,7H WATTS  /)
      DIMENSION X(2),YY(2),Y(325),CABS(4)
      COMPLEX INDX
      DATA (Y(I),I=1,71)  /
     1 0.0,.17E-7  ,  .136E-6  ,  .756E-6  ,  .317E-5  ,  .106E-4  ,
     2   .301E-4   ,  .738E-4  ,  .161E-3  ,  .321E-3  ,  .589E-3  ,
     3   .00131    ,  .00164   ,  .00252   ,  .00373   ,  .00531   ,
     4   .00733    ,  .00983   ,  .01285   ,  .01643   ,  .02060   ,
     5   .02537    ,  .03076   ,  .03677   ,  .04338   ,  .05059   ,
     6   .05838    ,  .06672   ,  .07559   ,  .08496   ,  .09478   ,
     7   .10503    ,  .11567   ,  .12665   ,  .13795   ,  .14953   ,
     8   .16135    ,  .17337   ,  .18556   ,  .19789   ,  .21033   ,
     9   .22285    ,  .23543   ,  .24803   ,  .26063   ,  .27322   ,
     A   .28576    ,  .29825   ,  .31067   ,  .32300   ,  .33523   ,
     B   .34734    ,  .35933   ,  .37118   ,  .38289   ,  .39445   ,
     C   .40586    ,  .41708   ,  .42815   ,  .43905   ,  .44977   ,
     D   .46031    ,  .47067   ,  .48085   ,  .49084   ,  .50066   ,
     E   .51029    ,  .51974   ,  .52901   ,  .53809   ,  .54700   /
      DATA (Y(I),I=72,141)  /
     F   .55573    ,  .56429   ,  .57267   ,  .58087   ,  .58891   ,
     G   .59678    ,  .60449   ,  .61203   ,  .61941   ,  .62664   ,
     H   .63371    ,  .64063   ,  .64740   ,  .65402   ,  .66051   ,
     I   .66685    ,  .67305   ,  .67912   ,  .68506   ,  .69087   ,
     J   .69655    ,  .70211   ,  .70754   ,  .71286   ,  .71806   ,
     K   .72315    ,  .72813   ,  .73301   ,  .73777   ,  .74244   ,
     L   .74700    ,  .75146   ,  .75583   ,  .76010   ,  .76429   ,
     M   .76838    ,  .77238   ,  .77630   ,  .78014   ,  .78390   ,
     N   .78757    ,  .79117   ,  .79469   ,  .79814   ,  .80152   ,
     O   .80482    ,  .80806   ,  .81123   ,  .81433   ,  .81737   ,
     P   .82035    ,  .82327   ,  .82612   ,  .82892   ,  .83166   ,
     Q   .83435    ,  .83698   ,  .83956   ,  .84209   ,  .84457   ,
     R   .84699    ,  .84937   ,  .85171   ,  .85399   ,  .85624   ,
     S   .85843    ,  .86059   ,  .86270   ,  .86477   ,  .86681   /
      DATA (Y(I),I=142,211)  /
     T   .86880    ,  .87075   ,  .87267   ,  .87455   ,  .87640   ,
     U   .87821    ,  .87999   ,  .88173   ,  .88344   ,  .88512   ,
     V   .88677    ,  .88839   ,  .88997   ,  .89153   ,  .89306   ,
     W   .89457    ,  .89604   ,  .89749   ,  .89891   ,  .90031   ,
     X   .90168    ,  .90303   ,  .90435   ,  .90565   ,  .90693   ,
     Y   .90819    ,  .90942   ,  .91063   ,  .91182   ,  .91299   ,
     Z   .91414    ,  .91527   ,  .91638   ,  .91748   ,  .91855   ,
     1   .91961    ,  .92064   ,  .92166   ,  .92267   ,  .92365   ,
     2   .92462    ,  .92558   ,  .92652   ,  .92744   ,  .92835   ,
     3   .92924    ,  .93012   ,  .93098   ,  .93183   ,  .93267   ,
     4   .93349    ,  .93510   ,  .93666   ,  .93816   ,  .93963   ,
     5   .94104    ,  .94242   ,  .94375   ,  .94504   ,  .94629   ,
```

```
     6  .94751    ,  .94869    ,  .94983    ,  .95094    ,  .95202    ,
     7  .95307    ,  .95409    ,  .95508    ,  .95604    ,  .95698    /
      DATA (Y(I),I=212,281)  /
     8  .95788    ,  .95877    ,  .95963    ,  .96046    ,  .96128    ,
     9  .96237    ,  .96284    ,  .96359    ,  .96432    ,  .96503    ,
     A  .96572    ,  .96639    ,  .96705    ,  .96769    ,  .96831    ,
     B  .96892    ,  .96951    ,  .97009    ,  .97065    ,  .97120    ,
     C  .97174    ,  .97226    ,  .97277    ,  .97327    ,  .97375    ,
     D  .97423    ,  .97469    ,  .97514    ,  .97558    ,  .97601    ,
     E  .97644    ,  .97685    ,  .97725    ,  .97764    ,  .97802    ,
     F  .97840    ,  .97877    ,  .97912    ,  .97947    ,  .97982    ,
     G  .98015    ,  .98048    ,  .98080    ,  .98111    ,  .98142    ,
     H  .98172    ,  .98201    ,  .98230    ,  .98258    ,  .98286    ,
     I  .98313    ,  .98339    ,  .98365    ,  .98390    ,  .98415    ,
     J  .98440    ,  .98463    ,  .98487    ,  .98510    ,  .98532    ,
     K  .98554    ,  .98576    ,  .98597    ,  .98617    ,  .98638    ,
     L  .98658    ,  .98677    ,  .98696    ,  .98715    ,  .98734    /
      DATA (Y(I),I=282,322)  /
     M  .98752    ,  .98769    ,  .98787    ,  .98804    ,  .98821    ,
     N  .98837    ,  .98853    ,  .98869    ,  .98885    ,  .98900    ,
     O  .98915    ,  .99051    ,  .99165    ,  .99262    ,  .99344    ,
     P  .99414    ,  .99475    ,  .99528    ,  .99574    ,  .99614    ,
     Q  .99649    ,  .99680    ,  .99707    ,  .99732    ,  .99754    ,
     R  .99773    ,  .99791    ,  .99806    ,  .99820    ,  .99833    ,
     S  .99845    ,  .99855    ,  .99865    ,  .99874    ,  .99882    ,
     T  .99889    ,  .99896    ,  .99902    ,  .99908    ,  .99913    ,
     U  .99918    /
      DATA   STFBLZ / 5.46E-12 /
      PI=.314159265E+01
      TWOPI = 2.*PI
      HALFPI = PI/2.
 2223 READ (5,899) IX,JX,IRGN,JRGN,NSTART,ISO,IPLUME,REX
      IF (IPLUME.EQ.4)                              RETURN
      IF (IX.EQ.0 .AND. JX.EQ.0)                    STOP
      IF (IPLUME.EQ.2)                              GO TO 995
      NRGN = IRGN*JRGN
      JRGN1=JRGN+1
      IRGN1=IRGN+1
      DO 37 J=1,JRGN
      READ (5,885) GAMMA
      GAMMAX(J) = GAMMA
      GAMMA = GAMMA * PI/180.
      TANG(J) = TAN(GAMMA)
      TANG2(J) = TANG(J)**2
   37 CGAMMA(J) = COS(GAMMA)
      HI = 1. / TANG(JRGN)
      DO 888 J=1,JRGN
      TNG2(J+1) = TANG2(J)
      TTNG(J+1)=TANG(J)
  888 TTNG2(J+1)=TTNG(J+1)**2
      TTNG(1)=0.0
      TNG2(1) = 0.0
      TTNG2(1)=0.0
C SUBSCRIPT K IDENTIFIES THE PROPERTY IN THE PROP(K,I,J) ARRAY
C K=1 PARTICLE NUMBER DENSITY (P/CM**3)
C K=2 PARTICLE TEMPERATURE (DEGREES KELVIN)
```

```
C K=3 PARTICLE RADIUS (MICRONS)
C K=4 PROJECTED AREA OF PARTICLES (MICRONS**2)
C K=5 GAS TEMPERATURE (DEGREES KELVIN)
C K=6 STATIC PRESSURE (PSIA)
C K=7  MOLE FRACTION FOR CO (-)
C K=8  MOLE FRACTION FOR CO2 (-)
C K=9  MOLE FRACTION FOR H2O (-)
C K=10 MOLE FRACTION FOR HCL (-)
      DO 882 I=1,IRGN1
      READ(5,885) HZ(I)
      HZ(I) = HZ(I) + H1
      DO 882 J=1,JRGN1
      READ (5,885) (PROP(K,I,J),K=1,6)
      READ (5,885) (PROP(K,I,J),K=7,10)
      PROP(1,I,J)=PROP(1,I,J)/28317. *1.0E9
      PROP(2,I,J)=(PROP(2,I,J)/1.8)**4
      PROP(3,I,J)=PROP(3,I,J)*30.48
      PROP(5,I,J)=PROP(5,I,J)/1.8
  882 PROP(5,I,J)=PROP(5,I,J)**4
      L=0
      DO 884 I=1,IRGN
      H2CUBE(I) = HZ(I)**3
      H1H2CB(I)=HZ(I+1)**3-H2CUBE(I)
      DO 884 J=1,JRGN
      L=L+1
      V(L) =
     1PI/3.0*(TTNG2(J+1)-TTNG2(J))*H1H2CB(I)
      DO 884 K=1,10
      SPJ  =(PROP(K,I+1,J)-PROP(K,I,J))/(HZ(I+1)-HZ(I))
      SPJ1=(PROP(K,I+1,J+1)-PROP(K,I,J+1))/(HZ(I+1)-HZ(I))
      DPJ=(PROP(K,I,J+1)-PROP(K,I,J))
      DSPJ =SPJ1-SPJ
      DTNJ   = TTNG(J+1)-TTNG(J)
      C1=DPJ/DTNJ
      C2=DSPJ/DTNJ
      A=PROP(K,I,J)-SPJ*HZ(I)-C1*TTNG(J)+C2*HZ(I)*TTNG(J)
      B=SPJ-C2*TTNG(J)
      C=C1-C2*HZ(I)
      D=C2
      A1=A*(TTNG2(J+1)-TTNG2(J))/2+C*(TTNG(J+1)**3-TTNG(J)**3)/3
      A2=B*(TTNG2(J+1)-TTNG2(J))/2+D*(TTNG(J+1)**3-TTNG(J)**3)/3
      P=TWOPI*(A1/3*(HZ(I+1)**3-HZ(I)**3)+A2/4*(HZ(I+1)**4-HZ(I)**4))
  884 PROP(K,I,J) = P/V(L)
      GO TO 996
  995 READ(5,885) PC,PAMB,TC,XK
      CALL QUICKP (PC,PAMB,TC,XK)
      H1=1./TANG(JRGN)
      NRGN = JRGN*IRGN
  996 CONTINUE
      XNAVG=0.0
      RADIAT = 0.0
      VT=0.0
      L=0
      XM=0.0
      XY=4.*STFBLZ*REX**3
      DO 886 I=1,IRGN
```

```
      DO 886 J=1,JRGN
      TG=PROP(5,I,J)**.25
      TP=PROP(2,I,J)**.25
      L=L+1
      VT=VT+V(L)
      CABS(1)      =(.3-.000225*(TG-1400.))/30.48
      IF(TG.GT.2500.)CABS(1)      =.05/30.48
      CABS(2)      =(4.4-.0048*(TG-1400.))/30.48
      IF(TG.GT.1800.)CABS(2)      =(2.4-.002*(TG-1800.))/30.48
      IF(TG.GT.2700.)CABS(2)      =.6/30.48
      CABS(3)      =(.8-.00054*(TG-1400.))/30.48
      IF(TG.GT.2500.)CABS(3)      =.2/30.48
      CABS(4)=CABS(3)
      XKA=0.0
      DO 34 K=1,4
      CABS(K)=CABS(K)*PROP(K+6,I,J)*PROP(6,I,J)/14.7
   34 XKA=XKA+CABS(K)
      TAUG(I,J) = REX*XKA
      PLANKA=0.0
      PLANKE=0.0
      W=0.0
      DO 688 N=1,40
      X(1)=TP*1.8*W
      X(2)=TP*1.8*(W+0.5)
      W=W+0.5
      DO 404 M=1,2
      IF(X(M).LE.1000.) GO TO 400
      IF(X(M).LE.20000.) GO TO 401
      IF(X(M).LE.40000.) GO TO 402
      IF(X(M).LE.100000.) GO TO 403
  400 II=1
      XC=X(M)/1000.
      GO TO 404
  401 II=X(M)/100.-8
      XC=(X(M)-1000.-(II-2)*100.)/100.
      GO TO 404
  402 II=X(M)/200.+92
      XC=(X(M)-20000.-(II-192)*200.)/200.
      GO TO 404
  403 II=X(M)/2000.+272
      XC=(X(M)-40000.-(II-292)*2000.)/2000.
  404 YY(M)= Y(II)+(Y(II+1)-Y(II))*XC
      IF(N.EQ.40)YY(2)=1.0
      BFP=YY(2)-YY(1)
      IF(TP.GT.2200.)INDX=CMPLX(1.8,.001)
      IF(TP.GE.2315.)INDX=CMPLX(1.8,.005)
      IF(TP.LT.2200.)INDX=CMPLX(1.8,.0001)
      CALL SPHERE(W     ,PROP(3,I,J),INDX ,QEXT,QABS)
      PLANKA=PLANKA+BFP*QABS
  688 PLANKE=PLANKE+BFP*QEXT
      QABS=PLANKA
      QEXT=PLANKE
      AE=QABS/QEXT
      TAUP(I,J)=PROP(1,I,J)*QEXT*PI*REX*PROP(3,I,J)**2*1.E-8
      IF(PROP(4,I,J).GT.0.0)
     A TAUP(I,J)=PROP(4,I,J)*PI*REX*QEXT/30.48
```

```
      TAU(I,J)=TAUG(I,J)+TAUP(I,J)
      CHIBTA(L)=(TAUG(I,J)+TAUP(I,J)*AE)/TAU(I,J)
      XNAVG=V(L)*PROP(1,I,J)*REX**3+XNAVG
      XNHU(L)=XY*V(L)*(AE*TAUP(I,J)/REX*PROP(2,I,J)+XKA*PROP(5,I,J))
      PROP(2,I,J)=TP
      PROP(5,I,J)=TG
      RADIAT=RADIAT+XNHU(L)
  886 XM=XM+PROP(1,I,J)*V(L)*PROP(3,I,J)**3
      VTT = VT * (REX/(12.*2.54))**3
      XNAVG=XNAVG/VTT
      XXNAVG = XNAVG/(12.*2.54)**3
      L=0
      DO 990 I=1,IRGN
      DO 990 J=1,JRGN
      L=L+1
      XNHU(L) =XNHU(L)/RADIAT
      IREMIT(L) = 0
      NEMIT(   L) = 0
      ISCAT(   I,J)=0
  990 CONTINUE
      RADIAT=RADIAT*FACTR
      WRITE(6,772) XNAVG,XXNAVG,VT ,REX , RADIAT
      DO 106 K=1,10
      WRITE(6,107) K
      DO 106 I = 1,IRGN
  106 WRITE(6,108) I,(PROP(K,I,J),J=1,JRGN)
  107 FORMAT (// 20H OUTPUT FOR PROPERTY  , I3 /)
  108 FORMAT (I3,2X,11E11.5)
      IF(ISU.EQ.0) RETURN
      READ (5,885) SA,SB,SC,SD,SE
      READ (5,885) SF,SG,SH,SI,SJ
      QA = SA
      QB = SB
      QC = SC
      QD = SD
      QE = SE
      SA=SA/180.0*PI
      SB=SB/180.0*PI
      SC=SC/180.0*PI
      SD=SD/180.0*PI
      SE=SE/180.0*PI
      A1=.5*SA*SJ
      A2=.5*(SB-SA)*(SJ-SH)
      A3=(SB-SA)*SH
      A4=.5*(SC-SB)*(SH-SG)
      A5=(SC-SB)*SG
      A6=.5*(SD-SC)*(SG-SF)
      A7=(SD-SC)*SF
      A8=.5*(SE-SD)*(SI-SF)
      A9=(SE-SD)*SF
      A10=.5*(PI-SE)*SI
      C1=A1+A2+A3
      C2=C1+A4+A5
      C3=C2+A6+A7
      C4=C3+A8+A9
      AK=C4+A10
```

```
      S1=2.0*AK*SA/SJ
      S2=(SB-SA)/(SJ-SH)
      S3=S2*S2*SJ*SJ
      S4=(SC-SB)/(SH-SG)
      S5=S4*S4*SH*SH
      S6=(SD-SC)/(SG-SF)
      S7=S6*S6*SG*SG
      S8=(SE-SD)/(SI-SF)
      S9=S8*S8*SF*SF
      C1K=C1/AK
      C2K=C2/AK
      C3K=C3/AK
      C4K=C4/AK
      A1K=A1/AK
      RETURN
      END

@PRT,S INTRCP
```

```
LEE-ALBIN202*TPFS.INTRCP
      SUBROUTINE INTRCP (X,A,KPLUME,KOUT,XP,AP)
C
C     X IS THE LOCATION, A IS THE VELOCITY D C
C     KPLUME IS THE ORIGINATING PLUME
C     KOUT EQ 0, IF NOT INTERCEPT ANY PLUME.
C     KOUT EQ 1 OR 2 IF WITHIN OR INTERCEPT THAT PLUME AT XP.
C
      INCLUDE DIM,LIST
      DIMENSION X(3),A(3),XP(3),AP(3),XPL(3),APL(3)
      KOUT=0
      KTARGT=1
      IF (KPLUME.EQ.1)                                  KTARGT=2
      CALL TRANSF (X,XPL,A,APL,0,KTARGT)
C     XPL AND APL ARE THE X AND A IN KTARGT PLUME LOCAL COORDINATES NOW.
      IRGN1=IRGN+1
      H1=1.0/TANG(JRGN)
      S=TANG(JRGN)
      HPL=HZ(IRGN1)
      IF (IPLUME.EQ.4)                                  GO TO 100
      HPL=HZ(IRGN1)-H1
  100 CONTINUE
      RLOC=(XPL(1)+H1)*TANG(JRGN)
      RPL=SQRT(XPL(2)**2+XPL(3)**2)
      IF (RPL.GT.RLOC)                                  GO TO 200
  120 KOUT=KTARGT
      DO 125 I=1,3
      XP(I)=XPL(I)
  125 AP(I)=APL(I)
      GO TO 300
C     CHECK FOR INTERCEPTION
  200 CONTINUE
      AA=1.-APL(2)**2 * (1.+S*S)
      B=XPL(2)*APL(2)+XPL(3)*APL(3)-S*S*(XPL(1)+H1)*APL(1)
      C=XPL(2)*XPL(2)+XPL(3)*XPL(3)-S*S*(XPL(1)+H1)*(XPL(1)+H1)
      CRIT=B*B-AA*C
      IF (CRIT.LT.0.)                                   GO TO 300
      D=(-B-SQRT(CRIT))/AA
      IF (D.LT.0.)                                      GO TO 300
      TZ=XPL(1)+D*APL(1)
      IF (TZ.LT.0..OR.TZ.GT.HPL)                        GO TO 300
      DO 250 I=1,3
      XP(I)=XPL(I)+D*APL(I)
  250 AP(I)=APL(I)
      PHI1=ATAN2(XP(3),XP(2))
      IF (PHI1.LT.0.)                                   PHI1=PHI1+TWOPI
      KOUT=KTARGT
  300 CONTINUE
      RETURN
      END

@PRT,S IOPKT
```

```
LLE-ALBIN202*TPFS.IOPKT
            SUBROUTINE IOPKT (M,IRMIT,X,R,E,T,P,IO)
            DIMENSION S(500)
      C *** IO=1, WRITE TAPE,  IO=2, READ TAPE.
            M=M+1
            GO TO (100,200), IO
        100 IF (M.EQ.1) MM=0
            I=5*MM
            S(I+1)=X
            S(I+2)=R
            S(I+3)=E
            S(I+4)=T
            S(I+5)=P
            MM=MM+1
            IF (MM.LT.100) GO TO 500
            WRITE (IO) IRMIT,(S(J), J=1,500)
            IF (M.EQ.100) WRITE (6,151)
            WRITE (6,150) M,MM,IRMIT,(S(J),J=496,500)
        150 FORMAT (2X,12HM MM IRMIT = ,3I10,5X,11HX R E T P = ,5F13.6 )
        151 FORMAT (//)
            MM=0
            GO TO 500
        200 CONTINUE
            IF (M.EQ.1) MM=0
            IF (MM.GT.0) GO TO 210
            READ  (IO) IRMIT,(S(J), J=1,500)
        210 I=5*MM
            X=S(I+1)
            R=S(I+2)
            E=S(I+3)
            T=S(I+4)
            P=S(I+5)
            MM=MM+1
            IF (MM.EQ.100) MM=0
        500 RETURN
            END

OPRT,S MAIN
```

```
LEE-ALBIN202*TPFS.MAIN
C *** MAIN PROGRAM FOR DUAL PLUME MONTE CARLO SIMULATION
C *** ICALC=0 OR 1 FOR CALCULATING HEATING RATE, =2 FOR VIEW FACTOR
      INCLUDE GEOM,LIST
      INCLUDE DIM,LIST
      COMMON /TWOPLM/PP(2,3),SIG(2),PSI(2)
      DIMENSION P(3),W(3),X(3),A(3),XP(3),AP(3)
      READ (5,3001) ICALC
      IF (ICALC.EQ.2)                                  GO TO 25
      READ (5,3002) ((PP(I,J),J=1,3),SIG(I),PSI(I), I=1,2)
      DO 20 I=1,2
      WRITE (6,3003) I,(PP(I,J),J=1,3),SIG(I),PSI(I)
      PSI(I)=PSI(I)/57.29578
   20 SIG(I)=SIG(I)/57.29578
   25 CALL TARGET
      IF (ICALC.EQ.2)                                  GO TO 2000
C *** CALCULATION OF HEATING RATES
      FACTR=2.0
   40 CALL INPUT (NRGN,HI,FACTR)
      H=0
      DO 50 K=1,3
      DO 50 J=1,NTMAX
      DO 50 I=1,NSIDEA
   50 HIT(I,J,K)=0.0
      CALL RNUM (NSTART)
      IF (IPLUME.EQ.3)                                 REWIND 10
      IF (IPLUME.EQ.4)                                 GO TO 500
      DO 400 IAX=1,IX
      DO 390 JXX=1,JX
  100 C LL RANDOM (U)
      KPLUME=1
      IF (U.GT.0.5)                                    KPLUME=2
      CALL RANDOM (U)
      XNU=0.
      CALL EMITT (NRGN,XNU,U,H,RIS,RI)
      GO TO 300
  200 IF (ISO.EQ.0)                                    GO TO 250
      CALL SCATTR
      GO TO 300
  250 CALL RANDOM (U)
      THETA=TWOPI*U
      CALL RANDOM (U)
      COSETA=1.-2.*U
      SINETA=SQRT(1.-COSETA*COSETA)
      TANETA=SINETA/COSETA
  300 CALL ZCOORD (RIS,RI,J,H,HI,JZ)
      CALL ATTEN(IXPR,RI,RIS,H,JZ,ZESCAP)
      IF (IXPR.EQ.3)                                   GO TO 200
      P(1)=H-HI
      P(2)=RI*COS(PHII)
      P(3)=RI*SIN(PHII)
      W(1)=COSETA
      W(2)=SINETA*COS(THETA)
      W(3)=SINETA*SIN(THETA)
      CALL TRANSF (P,X,W,A,KPLUME,D)
      CALL INTRCP (X,A,KPLUME,KOUT,XP,AP)
```

```
              IF (KOUT.EQ.0)                                          GO TO 380
              KPLUME=KOUT
              R1S=XP(2)*XP(2)+XP(3)*XP(3)
              R1=SQRT(R1S)
              H=XP(1)+H1
              IRGN1=IRGN+1
              IF (H.GT.HZ(IRGN1))                                     GO TO 100
              DO 350 IIX=1,IRGN1
              IF (H.GT.HZ(IIX))                                       GO TO 350
              INDEX=IIX-1
              JNDEX=JRGN
              GO TO 355
          350 CONTINUE
          355 CONTINUE
              GO TO 250
          380 CONTINUE
              IF (IPLUME.EQ.3)                                        GO TO 385
              GIMBLE=ABS(SIG(1))+ABS(SIG(2))+ABS(PSI(1))+ABS(PSI(2))
              IF (GIMBLE.GT.1.E-6 .OR. IXPR.EQ.1)                     GO TO 385
              M=M+1
              GO TO 390
          385 CALL ESCAP (X,A)
          390 CONTINUE
              IF (IPLUME.EQ.3)                                        GO TO 400
              CALL OUTPUT (H1,NRGN,0)
          400 CONTINUE
              GO TO 40
          500 CALL FRTAPE (H1,NRGN,2)
              GO TO 3000
       C *** CALCULATION OF VIEW FACTOR
         2000 CONTINUE
       C ***
              READ (5,3001) NRAY,NSTART
              IF (NRAY.LE.0)                                          GO TO 3000
              CALL VFEMIT (0,X,A,EAREA)
              M=0
              MMISS=0
              NHITP=0
              MHITG=0
              CALL RNUM (NSTART)
              DO 2050 K=1,3
              DO 2050 J=1,NTMAX
              DO 2050 I=1,NSIDEA
         2050 HIT(I,J,K)=0.0
       C
         2100 M=M+1
              IF (M.GT.NRAY)                                          GO TO 2500
       C
              CALL VFEMIT (1,X,A,EAREA)
       C
         2140 CALL SORTNG (X,A)
       C
              IF (ID.NE.0)                                            GO TO 2200
              MMISS=MMISS+1
              GO TO 2100
       C
```

```
      2200 CONTINUE
           IDX1=IDAREA
           IDX2=ID
           IDX3=IDISK+1
           HIT(IDX1,IDX2,IDX3) = HIT(IDX1,IDX2,IDX3)+1.0
           MHITG=MHITG+1
           GO TO 2100
     C
      2500 CONTINUE
           CALL VFOUTP (NRAY,NSTART,MHITP,MMISS,MHITG,EAREA)
           GO TO 2000
      3000 STOP
      3001 FORMAT (3I8)
      3002 FORMAT (10F8.0)
      3003 FORMAT (/ 10X,6HPLUME ,I2,10X,14HLOCATION PP = ,3F9.4,12X,
          1            7HSIGMA =,F7.2,6X,5HPSI = ,F7.2,8H DEGREES  /)
      3004 FORMAT (/10X,20HPLUME VERTEX ANGLE = ,F7.2,8H DEGREES ,8X,
          1            14HPLUME LENGTH = ,F7.2 // )
           END

@PRT,S MAINS
```

LEE-ALBIN202*TPFS.MAINS

```
               INCLUDE GEOM,LIST
               INCLUDE DIM,LIST
               DIMENSION IPLME(12),JPLME(12),KPLME(12),IZ(12),LTRU(12)
               CALL TARGET
               FACTR=1.0
          2223 CALL INPUT (NRGN,H1,FACTR)
           902 M=0
               DO 50 J=1,NTMAX
               DO 50 I=1,NSIDEA
               DO 50 K=1,3
            50 HIT(I,J,K)=0.0
               CALL RNUM(NSTART)
               IF (IPLUME.EQ.4)                                GO TO 3000
               IF (IPLUME.EQ.3)                                REWIND 10
               DO 81 I=1,12
               JPLME(I)=0
               KPLME(I)=0
               LTRU(I)=0
               IZ(I)=0
            81 IPLME(I)=0
               IOUT=0
               JOUT=0
               KOUT=0
               LOUT=0
               NHIT=0
               DPH=TWOPI/12.0
               DET=PI/12.0
               DO 2222 IXX = 1,IX
               DO 1112 JXX = 1,JX
         C DETERMINE REGION OF EMISSION POINT
             1 CALL RANDOM(U)
               X = 0
               CALL EMITT(NRGN,X,U,H,RIS,RI)
               GO TO 4
         C SELECT SCATTERING DIRECTION
             2 IF(ISO.EQ.0) GO TO 3
               CALL SCATTR
               GO TO 4
         C ISOTROPIC SCATTERING
             3 CALL RANDOM(U)
               THETA = TWOPI*U
               CALL RANDOM(U)
               COSETA = 1. - 2.*U
               SINETA = SQRT(1.-COSETA**2)
               TANETA = SINETA / COSETA
             4 CALL ZCOORD(RIS,RI,J,H,H1,JZ)
               CALL ATTEN(IXPR,RI,RIS,H,JZ,ZESCAP)
               IF(IXPR.EQ.3) GO TO 2
               CALL ESCAPE(H,H1,IXPR,RI,RIS,ZESCAP)
               IF (IU.NE.0)                                    NHIT=NHIT+1
            82 PH=0.0
               ET=0.0
               N=N+1
            85 CONTINUE
               ETA=ACOS(COSETA)
```

```
      ZS=ZESCAP-HI
      IF(ZS.LT.0.0) IZ(1)=IZ(1)+1
      IF(ZS.GE.0.0.AND.ZS.LT.2.0) IZ(2)=IZ(2)+1
      IF(ZS.GE.2.0.AND.ZS.LT.4.0) IZ(3)=IZ(3)+1
      IF(ZS.GE.4.0.AND.ZS.LT.6.0) IZ(4)=IZ(4)+1
      IF(ZS.GE.6.0.AND.ZS.LT.8.0) IZ(5)=IZ(5)+1
      IF(ZS.GE.8.0.AND.ZS.LT.10.) IZ(6)=IZ(6)+1
      IF(ZS.GE.10..AND.ZS.LT.12.) IZ(7)=IZ(7)+1
      IF(ZS.GE.12..AND.ZS.LT.14.) IZ(8)=IZ(8)+1
      IF(ZS.GE.14..AND.ZS.LT.16.) IZ(9)=IZ(9)+1
      IF(ZS.GE.16..AND.ZS.LT.18.) IZ(10)=IZ(10)+1
      IF(ZS.GE.18..AND.ZS.LT.20.) IZ(11)=IZ(11)+1
      IF(ZS.GE.20.0)              IZ(12)=IZ(12)+1
      T1=PH
      T2=ET
      TRUPH=THETA-PHI1
      IF (TRUPH.GT.TWOPI)                        TRUPH=TRUPH-TWOPI
      IF (TRUPH.LT.0.)                           TRUPH=TRUPH+TWOPI
      DO 86 I=1,12
      IF (PHI1.GT.T1 .AND. PHI1.LT.(T1+DPH))     IPLME(I)=IPLME(I)+1
      IF (ETA.GT.T2 .AND. ETA.LT.(T2+DET))       JPLME(I)=JPLME(I)+1
      IF (THETA.GT.T1 .AND. THETA.LT.(T1+DPH))   KPLME(I)=KPLME(I)+1
      IF (TRUPH.GT.T1 .AND. TRUPH.LT.(T1+DPH))   LTRU(I)=LTRU(I)+1
      T1=T1+DPH
   86 T2=T2+DET
      IF (PHI1.GT.TWOPI.OR.PHI1.LT.0.)          IOUT=IOUT+1
      IF (ETA.GT.PI)                             JOUT=JOUT+1
      IF (THETA.GT.TWOPI)                        KOUT=KOUT+1
      IF (TRUPH.GT.TWOPI)                        LOUT=LOUT+1
      IF (JXX.NE.JX)                             GO TO 1112
      NTRAJ=1
      DO 87 I=1,12
   87 NTRAJ=NTRAJ+IPLME(I)+JPLME(I)+KPLME(I)+LTRU(I)+IZ(I)
      NTRAJ=NTRAJ/5
      WRITE (6,89) (IPLME(I),I=1,12),(JPLME(J),J=1,12),(KPLME(I),I=1,12)
     1       ,(LTRU(I),I=1,12),(IZ(I),I=1,12),IOUT,JOUT,KOUT,LOUT,N,NTRAJ,
     2       NHIT
   89 FORMAT (////22H NO OF EMIT IN DEL PHI   ,1X,12I9 //
     1          1X,21HNO OF EMIT IN DEL ETA   ,1X,12I9 //
     2          1X,21HNO OT EMIT IN THETA     ,1X,12I9 //
     3          1X,21HNO OF EMIT IN TRU PHI   ,1X,12I9//
     4          1X,21HNO OF EMIT IN DEL  Z    ,1X,12I9 ////
     5          1X,34HIOUT JOUT KOUT LOUT N NTRAJ NHIT      , 7I9 //// )
 1112 CONTINUE
      IF (IPLUME.EQ.3)                           GO TO 2222
      CALL OUTPUT (HI,NRGN,0)
 2222 CONTINUE
      GO TO 2223
 3000 CALL FRTAPE (HI,NRGN,1)
 3001 STOP
      END
```

@PRT,S OUTPUT

```
LEL-ALBIN202*TPFS.OUTPUT
               SUBROUTINE OUTPUT (HI,NRGN,KODE)
               INCLUDE GEOM,LIST
               INCLUDE DIM,LIST
      C OUTPUT
            91 FORMAT(20X,2HA=,F7.4,3X,2HB=,F7.3,3X,2HC=,F7.3,3X,2HD=,F7.3,3X,
              12HE=,F7.3)
            94 FORMAT (4X,I3,6X,F9.1,I14,8X,1PE10.4,5X,E15.4,I13,7X,0PF10.6 )
            95 FORMAT (/22X,12HNN1  AVERAGE ,10X,1PE10.4,5X,E15.4 /)
           333 FORMAT (1H1)
           883 FORMAT (/4X,4HAREA,8X,9H NUMBER   ,8X,4HAREA,3X,2(3X,
              1          13HHEAT TRANSFER,4X),4X,4HAREA,8X,8HHIT AREA    /3X,
              2          6HNUMBER,6X,11H OF  HITS   ,6X,6HNUMBER,9X,5HW/CM2 ,
              3          12X,11HBTU/SEC-FT2,8X,6HNUMBER  )
           911 FORMAT(20X,2HF=,F7.4,3X,2HG=,F7.3,3X,2HH=,F7.3,3X,2HI=,F7.3,3X,
              12HJ=,F7.3/)
          1400 FORMAT( // 40H1             PROPERTIES IN PLUME REGIONS       // )
          1402 FORMAT(  //   26H TP - ( DEGREES KELVIN )     /
              1 26H TG - ( DEGREES KELVIN )          /
              1 20H N - ( PARTS/CM3 )    /
              - 15H TAUP - ( - )         /
              * 15H TAUG - ( - )         /
              2 15H TAU - ( - )          /
              3 20H RP - ( MICRONS )     /
              4 15H A/E - ( - )          /
              5 15H MF - ( - )           /
              7 30H HZ - ( EXIT RADII )                  /// )
          1403 FORMAT (10H   GAMMA =, 11F11.2 )
          1405 FORMAT (/I4,6H  TP =, 11E11.5   )
          8002 FORMAT ( I4,6H  TG =, 11E11.5   )
          1406 FORMAT ( I4,6H  N  =, 11E11.5   )
          1812 FORMAT ( I4,6H TAUP=, 11E11.5   )
          1813 FORMAT ( I4,6H TAUG=, 11E11.5   )
          1407 FORMAT ( I4,6H  TAU=, 11E11.5   )
          1408 FORMAT ( I4,6H  RP =, 11E11.5   )
          1409 FORMAT ( I4,6H  A/E=, 11E11.5   )
          1410 FORMAT ( I4,6HRADFK=, 11E11.5   )
          1412 FORMAT ( I4,6H  HZ =, 11E11.5 / )
          1600 FORMAT( // 60H DISTRIBUTION OF SCATTERINGS AND EMISSIONS THROUGHOU
              1T PLUME                      //)
          1601 FORMAT (6X,8H GAMMA =, 11F10.2 )
          1603 FORMAT (/I4,10H   NEMIT =, 11I10  )
          1604 FORMAT ( I4,10H   ISCAT =, 11I10  )
          1605 FORMAT ( I4,10H  ISCORE =, 11I10  )
          1606 FORMAT ( I4,10H   IMISS =, 11I10  )
          1607 FORMAT ( I4,10H REEMITS =, 11I10  )
          1900 FORMAT (38H1TOTAL NUMBER OF EMISSION/ABSORPTION =    ,I8,I10,7X,
              1            5HREX =,F7.1,7X,8HRADIAT =,1PE12.4,6H WATTS  /)
          1904 FORMAT (// 25H ANIOTROPIC SCATTERING    //)
          1906 FORMAT (// 25H ISOTROPIC SCATTERING     //)
          1909 FORMAT(11H TARGET NO.,I3)
        31     FORMAT(1X,27HON TARGET MAIN SIDE SURFACE)
        32     FORMAT(1X,34HON CONSTRAINT DISK PASSING THRU P1)
        33     FORMAT(1X,34HON CONSTRAINT DISK PASSING THRU P2)
               IF (IPLUME.EQ.4)                                 GO TO 100
               IF (NPPT.GE.1)                                   GO TO 100
```

```
                NPPT=NPPT+1
                WRITE(6,1400)
                WRITE (6,1402)
                WRITE( 6,1403 ) ( GAMMAX(J) , J=1,JRGN)
                K=0
                DO 1404 I= 1,IRGN
                WRITE(6,1405) I,(PROP(2,I,J),J=1,JRGN)
                WRITE(6,8002)I,(PROP(5,I,J),J=1,JRGN)
                WRITE(6,1406) I,(PROP(1,I,J),J=1,JRGN)
                WRITE(6,1812) I, (TAUP(I,J),J=1,JRGN)
                WRITE(6,1813) I, (TAUG(I,J),J=1,JRGN)
                WRITE (6,1407) I, (TAU(I,J), J=1,JRGN)
                WRITE(6,1408)I,(PROP(3,I,J),J=1,JRGN)
                WRITE( 6,1409) I , ( CHIBTA(K+J), J = 1,JRGN )
                WRITE(6,1410) I, (XNHU(  K+J), J=1,JRGN)
                HZ(I+1) = HZ(I+1) - H1
                WRITE(6,1412) I , HZ(I+1)
                HZ(I+1) = HZ(I+1) + H1
         1404 K=K+JRGN
      C --- OUTPUT START HERE
          100 CONTINUE
                IF( ISO . EQ . 0) GO TO 1903
                WRITE(6,1904)
                WRITE(6,91) QA,QB,QC,QD,QE
                WRITE(6,911)SF,SG,SH,SI,SJ
                GO TO 1905
         1903 WRITE(6,1906)
                IF (IPLUME.EQ.4)                               GO TO 200
         1905 WRITE(6,1600)
                WRITE( 6,1601 ) ( GAMMAX(J) , J=1,JRGN)
                L=1
                DO 1602  I = 1,IRGN
                LX=L+JRGN-1
                WRITE(6,1603) I , ( NEMIT(JX), JX=L,LX)
                WRITE(6,1607) I,(IREMIT(JX),JX=L,LX)
                L=LX+1
         1602 WRITE (6,1604) I, (ISCAT(I,J), J=1,JRGN)
          200 CONTINUE
      C FOR LOCAL THERMODYNAMIC EQUILIBRIUM
                IRMIT = 0
                DO 754 J = 1,NRGN
          754 IRMIT=IRMIT+IREMIT(J)
                XA = FLOAT(M+IRMIT)
      C
                DO 894 K=1,NTARGT
                IF (KODE.GT.0 .AND. K.NE.KODE)                 GO TO 894
                NCLOCK=DATA(K,1)
                DO 893 L=1,3
                IF (IBODY(K).GT.3.AND.L.NE.1) GO TO 893
                IF (IBODY(K).GT.2.AND.L.EQ.2) GO TO 893
                WRITE (6,1900) M,IRMIT,REX,RADIAT
                WRITE(6,1909) K
                GO TO (11,12,13),L
      11        WRITE(6,31)
                GO TO 15
      12        WRITE(6,32)
```

```
              GO TO 15
       13     WRITE(6,33)
       15     CONTINUE
              PRINT 883
              NA=NAREA(K,L)
              RGKNT=0.5
              RINGAV=0.
              DO 1688 J=1,NA
              GO TO (21,22,22),L
       21     HTAREA=VAREA(K,J)
              GO TO 25
       22     NCHECK=(J-1)/NCLOCK+1
              HTAREA=CRING(K,NCHECK,L-1)
         25   CONTINUE
              IF (HTAREA.LT.1.0E-8) GO TO 1688
              X=HIT(J,K,L)/M
              QN=RADIAT*HIT(J,K,L)/XA/HTAREA/REX/REX
              QNBTU=QN/1.1353
              WRITE (6,94) J,HIT(J,K,L),J,QN,QNBTU,J,HTAREA
              RGKNT=RGKNT+1.0
              RINGAV=RINGAV+QN
              IF (RGKNT.LT.DATA(K,1))                 GO TO 1688
              RINGAV=RINGAV/(RGKNT-0.5)
              RAVBTU=RINGAV/1.1353
              WRITE (6,95) RINGAV,RAVBTU
              RGKNT=0.5
              RINGAV=0.
       1688   CONTINUE
        893   CONTINUE
        894   CONTINUE
              RETURN
              END
```

```
@PRT,S PINGEA
```

```
LEE-ALBIN202*TPFS.PINGEA
              SUBROUTINE PINGEA (C,P,A,ROOTS,IPASS)
              DIMENSION C(10),P(3),A(3),ROOTS(4,2)
              RANGE(X1,Y1,Z1,X2,Y2,Z2)=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
              IPASS=1
              EPS=1.0E-10
              IF (ABS(A(1)).LT.EPS) GO TO 1
              W1=P(2)-P(1)*A(2)/A(1)
              W2=P(3)-P(1)*A(3)/A(1)
              A21=A(2)/A(1)
              A31=A(3)/A(1)
              Q1=C(1)+C(2)*A21**2+C(3)*A31**2+C(4)*A21+C(5)*A31+C(6)*A21*A31
              Q2=2.0*C(2)*A21*W1+2.0*C(3)*A31*W2+C(4)*W1+C(5)*W2+C(6)*(A31*W1
             *   +A21*W2)+C(7)+C(8)*A21+C(9)*A31
              Q3=C(2)*W1**2+C(3)*W2**2+C(6)*W1*W2+C(8)*W1+C(9)*W2+C(10)
              IF (ABS(Q1).LT.EPS) GO TO 121
              CALL QUADEQ(Q1,Q2,Q3,X1A,XIM1A,X1B,XIM1B,IFLAG)
              IF (IFLAG.EQ.3) GO TO 101
              GO TO 122
       121    IF (ABS(Q2).LT.EPS) GO TO 101
              X1A=-Q3/Q2
              X1B=X1A
       122    X2A=A21*X1A+W1
              X2B=A21*X1B+W1
              X3A=A31*X1A+W2
              X3B=A31*X1B+W2
              GO TO 102
       1      IF (ABS(A(2)).LT.EPS) GO TO 2
              A32=A(3)/A(2)
              W3=P(3)-P(2)*A32
              X1=P(1)
              X1A=X1
              X1B=X1
              Q1=C(2)+C(3)*A32**2+C(6)*A32
              Q2=2.0*C(3)*A32*W3+C(4)*X1+C(5)*A32*X1+C(6)*W3+C(8)+C(9)*A32
              Q3=C(1)*X1**2+C(3)*W3**2+C(5)*X1*W3+C(7)*X1+C(9)*W3+C(10)
              IF (ABS(Q1).LT.EPS) GO TO 123
              CALL QUADEQ(Q1,Q2,Q3,X2A,XIM2A,X2B,XIM2B,IFLAG)
              IF (IFLAG.EQ.3) GO TO 101
              GO TO 124
       123     IF (ABS(Q2).LT.EPS) GO TO 101
              X2A=-Q3/Q2
              X2B=X2A
       124    X3A=A32*X2A+W3
              X3B=A32*X2B+W3
              GO TO 102
       2      X1=P(1)
              X2=P(2)
              X1A=X1
              X1B=X1
              X2A=X2
              X2B=X2
              Q1=C(3)
              Q2=C(5)*X1+C(6)*X2+C(9)
              Q3=C(1)*X1**2+C(2)*X2**2+C(4)*X1*X2+C(7)*X1+C(8)*X2+C(10)
              IF (ABS(Q1).LT.EPS) GO TO 125
```

```
      CALL QUADEQ(Q1,Q2,Q3,X3A,XIM3A,X3B,XIM3B,IFLAG)
      IF (IFLAG.EQ.3) GO TO 101
      GO TO 102
125   IF (ABS(Q2).LT.EPS) GO TO 101
      X3A=-Q3/Q2
      X3B=X3A
102   CONTINUE
      SA=RANGE(P(1),P(2),P(3),X1A,X2A,X3A)
      SB=RANGE(P(1),P(2),P(3),X1B,X2B,X3B)
      ROOTS(1,1)=X1A
      ROOTS(2,1)=X2A
      ROOTS(3,1)=X3A
      ROOTS(4,1)=SA
      ROOTS(1,2)=X1B
      ROOTS(2,2)=X2B
      ROOTS(3,2)=X3B
      ROOTS(4,2)=SB
      GO TO 110
101   CONTINUE
      IPASS=0
110   RETURN
      END

@PRT,S PINGER
```

```
LEE-ALBIN202*TPFS.PINGEB
      SUBROUTINE PINGEB (C,P,A,SMIN,IPASS)
      DIMENSION C(10),P(3),A(3),SMIN(4)
      RANGE(X1,Y1,Z1,X2,Y2,Z2)=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
      IPASS=1
      EPS=1.0E-20
      CHECK1=C(7)*A(1)+C(8)*A(2)+C(9)*A(3)+C(10)
      IF (ABS(CHECK1).LT.EPS) GO TO 105
      IF (ABS(A(1)).LT.EPS) GO TO 1
      A21=A(2)/A(1)
      A31=A(3)/A(1)
      W1=P(2)-A21*P(1)
      W2=P(3)-A31*P(1)
      X1=-(C(8)*W1+C(9)*W2+C(10))/(C(7)+C(8)*A21+C(9)*A31)
      X2=A21*X1+W1
      X3=A31*X1+W2
      GO TO 102
    1 IF (ABS(A(2)).LT.EPS) GO TO 2
      A32=A(3)/A(2)
      W3=P(3)-A32*P(2)
      X1=P(1)
      X2=-(C(7)*X1+C(9)*W3+C(10))/(C(8)+C(9)*A32)
      X3=A32*X2+W3
      GO TO 102
    2 X1=P(1)
      X2=P(2)
      X3=-(C(7)*X1+C(8)*X2+C(10))/C(9)
  102 CONTINUE
      SMIN(1)=X1
      SMIN(2)=X2
      SMIN(3)=X3
      SMIN(4)=RANGE(P(1),P(2),P(3),X1,X2,X3)
      GO TO 110
  105 IPASS=0
      CHECK2=C(7)*P(1)+C(8)*P(2)+C(9)*P(3)+C(10)
      IF (ABS(CHECK2).LT.EPS) GO TO 106
      WRITE(6,22)
      GO TO 110
  106 WRITE(6,23)
      IPASS=1
      SMIN(1)=P(1)
      SMIN(2)=P(2)
      SMIN(3)=P(3)
      SMIN(4)=0.0
  110 RETURN
   22 FORMAT   (2X,43HTHE STRAIGHT LINE IS PARALLEL TO THE PLANE.)
   23 FORMAT   (2X,36HTHE STRAIGHT LINE LIES IN THE PLANE.)
      END

@PRT,S QUADEQ
```

```
LEE-ALBIN202*TPFS.QUADEQ
      SUBROUTINE QUADEQ (A,B,C,XR1,XIM1,XR2,XIM2,IFLAG)
      IWRITE=0
      DISC=B**2-4.0*A*C
      IF (DISC) 50,60,70
   50 IFLAG=3
      XR1=-B/(2.0*A)
      XR2=XR1
      XIM1=SQRT(-DISC)/(2.0*A)
      XIM2=-XIM1
      GO TO 100
   60 IFLAG=2
      XR1=-B/(2.0*A)
      XR2=XR1
      XIM1=0.0
      XIM2=0.0
      GO TO 100
   70 IFLAG=1
      S=SQRT(DISC)
      XR1=(-B+S)/(2.0*A)
      XR2=(-B-S)/(2.0*A)
      XIM1=0.0
      XIM2=0.0
  100 CONTINUE
      IF(IWRITE.EQ.0) GO TO 5
      WRITE(6,11) A,B,C
      GO TO (1,2,3),IFLAG
    1 WRITE(6,12) XR1,XIM1,XR2,XIM2
      GO TO 5
    2 WRITE(6,13) XR1,XIM1,XR2,XIM2
      GO TO 5
    3 WRITE(6,14) XR1,XIM1,XR2,XIM2
    5 RETURN
   11 FORMAT  (/,2X,33HCOEFFICENTS OF QUADRATIC EQUATION,8X,2HA=,E12.5,
     * 8X,2HB=,E12.5,8X,2HC=,E12.5)
   12 FORMAT(10X,28HUNEQUAL REAL ROOTS --- X1= (,E12.5,8H ) + I (,E12.5,
     *2H ),/,33X,5HX2= (,E12.5,8H ) + I (,E12.5,2H ))
   13 FORMAT(10X,28HEQUAL  REAL  ROOTS --- X1= (,E12.5,8H ) + I (,E12.5,
     *2H ),/,33X,5HX2= (,E12.5,8H ) + I (,E12.5,2H ))
   14 FORMAT(10X,28HCOMPLEX    ROOTS    --- X1= (,E12.5,8H ) + I (,E12.5,
     *2H ),/,33X,5HX2= (,E12.5,8H ) + I (,E12.5,2H ))
      END

@PRT,S QUICKP
```

```
LEE-ALBIN202*TPFS.QUICKP
              SUBROUTINE QUICKP (PC,PAMB,TC,XK)
              INCLUDE DIM,LIST
              DIMENSION      C2(5),C3(5),      XNN(5),T(5)
              DATA  C2(1),C2(2),C2(3),C2(4),C2(5)/
             1 -72.,-72.,-72.,-105.,-105.
             2 /
              DATA  C3(1),C3(2),C3(3),C3(4),C3(5)/
             1 1.05,1.05,1.00,.95,.9      /
              DATA  T(1),T(2),T(3 ),T(4),T(5)/
             1 2300.,2250.,2200.,2150.,2100./
              IRGN = 9
              JRGN = 5
              NRGN = IRGN*JRGN
              IRGN1 = IRGN + 1
              JRGN1 = JRGN + 1
              XN = PC * 20.
              PK = PC / PAMB
              XNN(1) = .95*XN
              XNN(2) = 1.1*XN
              XNN(3) = XN
              XNN(4) = .9 *XN
              XNN(5) = .8*XN
              IF(PK.LE.600.)GGAMMA = 2. + .025* PK
              IF(PK.GT.600.)GGAMMA = 17. + .001 * PK
              GGAMMA=GGAMMA/5.
              GAMMA=0.0
              DO 101 J=1,5
              GAMMA = GAMMA + GGAMMA
              GAMMAX(J) = GAMMA
              G = GAMMA*PI/180.
              CGAMMA(J) = COS(G)
              TANG(J) = TAN(G)
              TANG2(J) = TANG(J)**2
              TNG2(J+1) = TANG2(J)
              TTNG(J+1) = TANG(J)
          101 TTNG2(J+1) = TTNG(J+1)**2
              TTNG(1) = 0.0
              TNG2(1) = 0.0
              TTNG2(1)= 0.0
              H1 = 1./TANG(5)
              HZ(1) = 0.0
              HZ(2) = 0.2
              HZ(3) = C.4
              HZ(4) = 0.6
              HZ(5) = 1.0
              HZ(6) = 2.0
              HZ(7) = 4.0
              HZ(8) = 6.0
              HZ(9) = 8.0
              HZ(10) = 10.0
              DO 102 I=1,10
          102 HZ(I) = HZ(I)+H1
              L=0
              DO 104 I = 1,IRGN
              H2CUBE(I) = HZ(I)**3
```

```
      H1H2CB(I) = HZ(I+1)**3 - H2CUBE(I)
      DO 104 J = 1,JRGN
      L=L+1
      V(L) = PI/3.0 *(TTNG2(J+1)-TTNG2(J))*H1H2CB(I)
      X= PI*H1**2*(TTNG2(J+1)-TTNG2(J))*(HZ(I+1)-HZ(I))/V(L)
      PROP(1,I,J) = XNN(J)*X
      PROP(2,I,J) = 2317.
      H=HZ(I+1)-H1
      IF(H.GT..5.AND.H.LT.4.) PROP(2,I,J) = 2317.+C2(J)*(H-.4)
      IF(H.GE.4.) PROP(2,I,J) = C3(J)*2000.
      PROP(3,I,J)=6.0
      PROP(3,I,5)=3.0
      PROP(5,I,J)=T(J)*X**(XK-1)
      PROP(6,I,J)=PC*X/8.0*PROP(5,I,J)/TC
      IF(PROP(6,I,J).LT.PAMB)PROP(6,I,J)=PAMB
      PROP(7,I,J)=.249
      PROP(8,I,J)=.026
      PROP(9,I,J)=.152
      PROP(10,I,J)=.161
      PROP(2,I,J)=PROP(2,I,J)**4
  104 PROP(5,I,J)=PROP(5,I,J)**4
      RETURN
      END
```

QPRT,S SCATTR

```
LEE-ALBIN202*TPFS.SCATTR
            SUBROUTINE SCATTR
            INCLUDE DIM,LIST
      C ANISOTROPIC SCATTERING
        904 CALL RANDOM(U)
            IF(U-A1K)700,701,702
        700 ETAD=SQRT(S1*U)
            GO TO 905
        701 ETAD=SA
            GO TO 905
        702 IF(U-C1K)703,704,705
        703 PM213=(S3-2.0*(U*AK-A1)*S2)
            IF(PM213.LT.0.0) PM213=0.0
            ETAD=SA+S2*SJ-SQRT(PM213)
            GO TO 905
        704 ETAD=SB
            GO TO 905
        705 IF(U-C2K)706,707,708
        706 PM213=(S5-2.0*(U*AK-C1)*S4)
            IF(PM213.LT.0.0) PM213=0.0
            ETAD=SB+S4*SH-SQRT(PM213)
            GO TO 905
        707 ETAD=SC
            GO TO 905
        708 IF(U-C3K)709,710,711
        709 PM213=(S7-2.0*(U*AK-C2)*S6)
            IF(PM213.LT.0.0) PM213=0.0
            ETAD=SC+S6*SG-SQRT(PM213)
            GO TO 905
        710 ETAD=SD
            GO TO 905
        711 IF(U-C4K)712,713,714
        712 PM213=(S9+2.0*(U*AK-C3)*S8)
            IF(PM213.LT.0.0) PM213=0.0
            ETAD=SD-S8*SF+SQRT(PM213)
            GO TO 905
        713 ETAD=SE
            GO TO 905
        714 ETAD=PI-SQRT(AK*(1.0-U)*2.0*(PI-SE)/SI)
        905 CALL RANDOM(U)
            THETD = TWOPI * U
            COETD=COS(ETAD)
            SIETD=SIN(ETAD)
            COTHD=COS(THETD)
            SITHD=SIN(THETD)
            SITH=SIN(THETA)
            COTH=COS(THETA)
            ROX=SIETD*COTHD
            REY=SIETD*SITHD
            REZ=COETD
            RX= ROX*SITH+REY*COSETA*COTH+REZ*SINETA*COTH
            RY=-ROX*COTH+REY*COSETA*SITH+REZ*SINETA*SITH
            RZ=-REY*SINETA+REZ*COSETA
            THETA=ATAN(RY/RX)
            IF(RX)112,113,113
        112 THETA=THETA+PI
```

```
          GO TO 115
      113 IF(THETA)114,115,115
      114 THETA=THETA+TWOPI
      115 COSETA = RZ
          SINETA = SQRT(1.-COSETA**2)
          TANETA = SINETA / COSETA
          COST = COS(THETA)
          RETURN
          END
```

```
@PRT,S SORTNG
```

```
LEE-ALBIN202*TPFS.SORTNG
                 SUBROUTINE SORTNG (P,A)
                 INCLUDE GEOM,LIST
                 COMMON/RESULT/ ID,IDISK,IDAREA,KRING,KPHI,DISTNS,XX(3)
                 DIMENSION C(10),P(3),A(3),SMIN(4)
                 DIMENSION P1(3),P2(3),ROOTS(4,2)
                 RANGE(X1,Y1,Z1,X2,Y2,Z2)=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
                 ATRIG(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3)=
                *  SQRT(  (X1*(Y2-Y3)+X2*(Y3-Y1)+X3*(Y1-Y2))**2
                *            +(Z1*(X2-X3)+Z2*(X3-X1)+Z3*(X1-X2))**2
                *            +(Y1*(Z2-Z3)+Y2*(Z3-Z1)+Y3*(Z1-Z2))**2   )/2.0
                 IWRITE=0
                 PI=3.14159265
                 EPS=1.0E-4
                 DISTNS=1.0E30
                 ID=0
                 DO 900 NDATA=1,NTARGT
                 IDI=0
                 DO 500 II=1,10
         500     C(II)=COEF(NDATA,II)
                 IF (IBODY(NDATA).EQ.5)                    GO TO 401
                 IF (IBODY(NDATA).EQ.6)                    GO TO 402
                 CALL PINGEA(C,P,A,ROOTS,IPASS)
                 IF (IPASS.EQ.0) GO TO 400
                 ICHOSE=IBODY(NDATA)
                 DO 601 II=1,3
                 P1(II)=DATA(NDATA,II+4)
         601     P2(II)=DATA(NDATA,II+7)
                 IF (ICHOSE.LT.4) GO TO 404
                 DO 405 II=1,3
         405     P1(II)=2.0*P1(II)-P2(II)
         404     CALL CHOSE(P1,P2,ROOTS,SMIN,IOUT)
                 IF(IOUT.NE.0) GO TO 400
                 GO TO 402
         401     CALL PINGEB(C,P,A,SMIN,IPASS)
                 IF (IPASS.EQ.1) GO TO 402
                 GO TO 400
         402     CONTINUE
                 IF (IBODY(NDATA).NE.6)                    GO TO 403
                 CALL DISK (1,NDATA,P,A,SMIN,IHIT)
                 GO TO 412
          403    CONTINUE
                 CHECK1=SMIN(4)*A(1)-(SMIN(1)-P(1))
                 CHECK2=SMIN(4)*A(2)-(SMIN(2)-P(2))
                 CHECK3=SMIN(4)*A(3)-(SMIN(3)-P(3))
                 SUMCHK=ABS(CHECK1)+ABS(CHECK2)+ABS(CHECK3)
                 IF (SUMCHK.GT.EPS)                        GO TO 400
                 IF (IDI.NE.0) GO TO 411
                 IF (IBODY(NDATA).NE.5) GO TO 412
                 PLA1=ATRIG(  SMIN(1),SMIN(2),SMIN(3),
                *   DATA(NDATA,5),DATA(NDATA,6),DATA(NDATA,7),
                *   DATA(NDATA,8),DATA(NDATA,9),DATA(NDATA,10)   )
                 PLA2=ATRIG(  SMIN(1),SMIN(2),SMIN(3),
                *   DATA(NDATA,8),DATA(NDATA,9),DATA(NDATA,10),
                *   DATA(NDATA,11),DATA(NDATA,12),DATA(NDATA,13)   )
                 PLA3=ATRIG(  SMIN(1),SMIN(2),SMIN(3),
```

```
           •    DATA(NDATA,11),DATA(NDATA,12),DATA(NDATA,13),
           •    DATA(NDATA,14),DATA(NDATA,15),DATA(NDATA,16)   )
                PLA4=ATRIG(   SMIN(1),SMIN(2),SMIN(3),
           •    DATA(NDATA,5),DATA(NDATA,6),DATA(NDATA,7),
           •    DATA(NDATA,14),DATA(NDATA,15),DATA(NDATA,16)   )
                PCHECK=PLAREA(NDATA)-(PLA1+PLA2+PLA3+PLA4)
                IF (ABS(PCHECK/PLAREA(NDATA)).GT.EPS) GO TO 400
                GO TO 412
      411       IF (IBODY(NDATA).GT.3) GO TO 400
                IF (ID1.EQ.1 .AND. IBODY(NDATA).LE.2)          GO TO 413
                IF (ID1.EQ.2) GO TO 415
                GO TO 400
      415       HH1=RANGE(SMIN(1),SMIN(2),SMIN(3),DATA(NDATA,8),DATA(NDATA,9),
           •    DATA(NDATA,10))
                GO TO 414
      413       HH1=RANGE(SMIN(1),SMIN(2),SMIN(3),DATA(NDATA,5),DATA(NDATA,6),
           •    DATA(NDATA,7))
      414       RCHECK=RBOND(NDATA)
                IF (IBODY(NDATA).EQ.2.AND.ID1.EQ.1)          RCHECK=RBONDU(NDATA)
                IF (HH1.GT.RCHECK.OR.HH1.LT.RRING(NDATA,ID1)) GO TO 400
      412       CONTINUE
                IF (DISTNS.LT.SMIN(4)) GO TO 400
                DISTNS=SMIN(4)
                ID=NDATA
                IDISK=ID1
                IF (IBODY(NDATA).EQ.6)                        IDAREA=IHIT
                XX(1)=SMIN(1)
                XX(2)=SMIN(2)
                XX(3)=SMIN(3)
      400       CONTINUE
                IF (IBODY(NDATA).GT.3) GO TO 900
      450       ID1=ID1+1
                IF (ID1.GT.2) GO TO 900
                IF (IBODY(NDATA).GT.2 .AND. ID1.EQ.1)          GO TO 450
                DO 421 II=1,6
      421       C(II)=0.0
                C(7)=DISKEQ(NDATA,1)
                C(8)=DISKEQ(NDATA,2)
                C(9)=DISKEQ(NDATA,3)
                GO TO (431,432),ID1
      431       C(10)=DISKEQ(NDATA,4)
                GO TO 401
      432       C(10)=DISKEQ(NDATA,5)
                GO TO 401
      900       CONTINUE
                IF (IWRITE.EQ.0) GO TO 302
                WRITE(6,5) P(1),P(2),P(3),A(1),A(2),A(3)
                IF (ID.EQ.0) GO TO 301
                WRITE(6,7) ID,IDISK,DISTNS,XX(1),XX(2),XX(3)
                GO TO 302
      301       WRITE(6,8)
      302       CONTINUE
                IF (IBODY(ID).EQ.6)                            GO TO 1000
                IDAREA=0
                IF (ID.EQ.0) GO TO 1000
                IF (IDISK.NE.0) GO TO 1100
```

```
      NTOTAL=NAREA(ID,1)
      SS=1.0E30
      DO 110 II=1,NTOTAL
      TVIEW=RANGE(XX(1),XX(2),XX(3),VECTOR(ID,II,1),VECTOR(ID,II,2),
     * VECTOR(ID,II,3))
      IF (TVIEW.GE.SS) GO TO 110
      SS=TVIEW
      IDAREA=II
  110 CONTINUE
      NARC=DATA(ID,1)
      NFRUST=DATA(ID,2)
      IF (IBODY(ID).NE.4) GO TO 101
      P2P1=RANGE(DATA(ID,8),DATA(ID,9),DATA(ID,10),
     *            DATA(ID,5),DATA(ID,6),DATA(ID,7))
      P1T=RANGE(DATA(ID,5),DATA(ID,6),DATA(ID,7),XX(1),XX(2),XX(3))
      DOT=(DATA(ID,5)-DATA(ID,8))*(XX(1)-DATA(ID,5))
     *   +(DATA(ID,6)-DATA(ID,9))*(XX(2)-DATA(ID,6))
     *   +(DATA(ID,7)-DATA(ID,10))*(XX(3)-DATA(ID,7))
      COSGM=DOT/(P2P1*P1T)
      GAMMA=ACOS(COSGM)
  102 PHI=PI/DATA(ID,2)
      RTOP=NFRUST
      ANGTOP=RTOP*PHI
      IF (GAMMA.GT.ANGTOP) GO TO 105
      NF=(IDAREA-1)/NARC+1
      RN=NF
      PHIRN1=PHI*(RN-1)
      PHIRN2=PHI*RN
      IF (GAMMA.GE.PHIRN1.AND.GAMMA.LT.PHIRN2) GO TO 105
      IF (GAMMA.LT.PHIRN1) GO TO 103
      NF=NF+1
      IDAREA=IDAREA+NARC
      GO TO 105
  103 NF=NF-1
      IDAREA=IDAREA-NARC
      GO TO 105
  101 CONTINUE
      NF=(IDAREA-1)/NARC+1
      IF (IBODY(ID).LE.2 .OR. IBODY(ID).GE.5)      GO TO 106
      P1T=RANGE(DATA(ID,5),DATA(ID,6),DATA(ID,7),XX(1),XX(2),XX(3))
      P1A=RANGE(DATA(ID,5),DATA(ID,6),DATA(ID,7),VECTOR(ID,IDAREA,1),
     * VECTOR(ID,IDAREA,2),VECTOR(ID,IDAREA,3))
      IF (P1T.LT.P1A) GO TO 191
      IF (IDAREA.LE.NARC) GO TO 106
      IP1=IDAREA-NARC
      P1B=RANGE(DATA(ID,5),DATA(ID,6),DATA(ID,7),VECTOR(ID,IP1,1),
     * VECTOR(ID,IP1,2),VECTOR(ID,IP1,3))
      P1AB=(P1A+P1B)/2.0
      IF (P1T.LE.P1AB) GO TO 106
      NF=NF-1
      IDAREA=IP1
      GO TO 106
  191 IF (IDAREA.GT.NTOTAL-NARC) GO TO 106
      IP1=IDAREA+NARC
      P1B=RANGE(DATA(ID,5),DATA(ID,6),DATA(ID,7),VECTOR(ID,IP1,1),
     * VECTOR(ID,IP1,2),VECTOR(ID,IP1,3))
```

```
                PIAB=(PIA+PIB)/2.0
                IF (PIT.GE.PIAB) GO TO 106
                NF=NF+1
                IDAREA=IP1
        106     CONTINUE
        105     IF(IWRITE.EQ.0) GO TO 1000
                WRITE(6,121) IDAREA,NAREA(ID,1),NF,VAREA(ID,IDAREA)
                GO TO 1000
        1100    CONTINUE
                IF (IDISK.EQ.2) GO TO 1200
                PRX=XX(1)-DATA(ID,5)
                PRY=XX(2)-DATA(ID,6)
                PRZ=XX(3)-DATA(ID,7)
                GO TO 1201
        1200    PRX=XX(1)-DATA(ID,8)
                PRY=XX(2)-DATA(ID,9)
                PRZ=XX(3)-DATA(ID,10)
        1201    NRG=NCRING(ID,IDISK)
                PRMAG=RANGE(PRX,PRY,PRZ,0.0,0.0,0.0)
                IF (NRG.LE.1) GO TO 1104
                DO 1101 KK=2,NRG
                RAVER=(RMEANC(ID,KK,IDISK)+RMEANC(ID,KK-1,IDISK))/2.0
                IF (PRMAG.LT.RAVER) GO TO 1102
        1101    CONTINUE
                KRING=NRG
                GO TO 1103
        1102    KRING=KK-1
                GO TO 1103
        1104    KRING=1
        1103    SIGNT=PRX*TREF(ID,1,IDISK)+PRY*TREF(ID,2,IDISK)
               *     +PRZ*TREF(ID,3,IDISK)
                SIGNS=PRX*SREF(ID,1,IDISK)+PRY*SREF(ID,2,IDISK)
               *     +PRZ*SREF(ID,3,IDISK)
                ALNGT=RANGE(TREF(ID,1,IDISK),TREF(ID,2,IDISK),TREF(ID,3,IDISK),
               * 0.0,0.0,0.0)
                DENOM=ALNGT*PRMAG
                IF (ABS(DENOM).LT.1.0E-10) GO TO 1302
                ROTANG=SIGNT/DENOM
                IF (ABS(ROTANG).GT.1.0) ROTANG=SIGN(1.0,ROTANG)
                CHECK=ABS(ACOS(ROTANG))
                IF (SIGNT.GE.0.0.AND.SIGNS.GE.0.0) GO TO 1301
                IF (SIGNT.LE.0.0.AND.SIGNS.GE.0.0) GO TO 1301
                ANG=2.0*PI-CHECK
                GO TO 1310
        1301    ANG=CHECK
                GO TO 1310
        1302    ANG=0.0
        1310    COUNT=ANG/DPHIC(ID)
                IID=DATA(ID,1)
                KPHI=COUNT+1.0
                IF (KPHI.GT.IID) KPHI=IID
                ANG1=ANG*180.0/PI
                IDAREA=(KRING-1)*IID+KPHI
                IF (IWRITE.EQ.0) GO TO 1000
                WRITE(6,1311) KRING,KPHI,ANG1,CRING(ID,KRING,IDISK),IDAREA
        1000    RETURN
```

```
    1     FORMAT(I5)
    2     FORMAT(8E10.5)
    5     FORMAT(/,1X,14HEMISSION POINT,2X,6HP(X1)=,E11.5,2X,6HP(X2)=,E11.5,
         * 2X,6HP(X3)=,E11.5,2X,17HDIRECTION COSINES,3E12.5)
    7     FORMAT(3X,14HHIT TARGET NO.,I2,4X,6HDISK=,I2,4X,
         * 9HDISTANCE=,E11.5,5X,3HX1=,E11.5,3X,3HX2=,E11.5,3X,3HX3=,E11.5)
    8     FORMAT(2X,14HNO IMPINGEMENT)
  121      FORMAT(3X,11HAREA INDEX=,I3,3X,22HOF TOTAL NO. OF AREAS=,I3,5X,
         *28HFRUSTUM (AXIS DIRECTION) NO.,I3,3X,17HVIEW POINT AREA =,E10.4)
 1311      FORMAT(3X,24HCONSTRAINT DISK RING NO.,I3,3X,7HPHI NO.,
         * I3,3X,18HANGLE(CLOCKWISE) =,F8.3,3X,17HVIEW POINT AREA =,E10.4,
         * 3X,11HAREA INDEX=,I3)
          END

@PRT,S SPHERE
```

```
LEE*ALBIN202*TPFS.SPHERE
      SUBROUTINE SPHERE(WAVE,RADIUS,INDEX,QEXT,QABS)
      DIMENSION A(120,2),B(120,2)
      EQUIVALENCE (RARG,XKA)                                            SNGL0040
      DATA KONTRL,LENGTH/1,0/                                           SNGL005A
      DATA LL /12/                                                      SNGL0060
      REAL                                                              SNGL0090
     A    AI    ,  AIS   ,  AR    ,  AREA  ,  AREA1 ,  ARS   ,          SNGL0100
     B    A1    ,  A2    ,  B1    ,  BIS   ,  BR    ,  BRS   ,          SNGL0110
     C    CCAB  ,  CEXT  ,  CHIZER,  CI    ,  CONST ,  CR    ,          SNGL0120
     D    CRPR  ,  CSCA  ,  DEN   ,  DI    ,  DR    ,  DXR   ,          SNGL0130
     E    DYI   ,  DYR   ,  D10P6 ,  ENI   ,  ENR   ,  EXT   ,          SNGL0140
     F    FAC   ,  FACTOR,  FI    ,  FIFTY ,  FOU   ,  FR    ,          SNGL0150
     G    GREAT ,  HALF  ,           ONE   ,  PI    ,  RATIO ,          SNGL0160
     H    RPR   ,  RX    ,  RXN   ,  RX2   ,  RYI   ,  RYR   ,          SNGL0170
     I    SCA   ,  SIGI  ,  SIGR  ,  SMALL ,  THR   ,  TI    ,          SNGL0180
     J    TIGN  ,  TR    ,  TWO   ,  TWOPI ,  X     ,  XL    ,          SNGL0190
     K    YI    ,  YR    ,  Z     ,  ZER   ,  DARG  ,  DBLEF ,          SNGL0200
     L    PSI   ,  CHI   ,  GR    ,  GI    ,  A     ,  B                SNGL0210
      DATA NN/120/                                                      SNGL0270
      EQUIVALENCE (N,NUMBER)                                            SNGL0290
      DIMENSION CHI(1),PSI(1),GR(1),GI(1)                               SNGL0300
      EQUIVALENCE (A(1,1),CHI(1)),(A(1,2),PSI(1)),                      SNGL0310
     1            (B(1,1), GR(1)),(B(1,2), GI(1))                       SNGL0320
      COMPLEX INDEX,JNDEX,CARG                                          SNGL0410
      EQUIVALENCE (JNDEX,CARG),(X,DARG)                                 SNGL0420
      REAL SNGLF                                                        SNGL0430
      DATA ZER,ONE,TWO,THR,FOU,HALF/0.0E0,1.0E0,2.0E0,3.0E0,4.0E0,0.5E0/SNGL0450
      DATA PI,TWOPI /3.14159265,6.28318508/                             SNGL0460
      DATA TIGN /1.0E0/                                                 SNGL0470
      DATA SMALL,GREAT,FIFTY,D10P6/1.0E-35,1.0E+35,50.0E0,1.0E+6/       SNGL0480
      SNGLF(DARG)  = DARG                                               SNGL0490
      SQRTF(DARG)  = SQRT(DARG)                                         SNGL0500
      REALF(CARG)  = REAL(CARG)                                         SNGL0510
      AIMAF(CARG)  = AIMAG(CARG)                                        SNGL0520
      EXPF(DARG)   = EXP(DARG)                                          SNGL0530
      COSF(DARG)   = COS(DARG)                                          SNGL0540
      SINF(DARG)   = SIN(DARG)                                          SNGL0550
      IFIXF(DARG)  = IFIX(DARG)                                         SNGL0560
      ALOGF(DARG)  = ALOG(DARG)                                         SNGL0570
      ABSF(DARG)   = ABS(DARG)                                          SNGL0580
      FLOATF(I)    = FLOAT(I)                                           SNGL0590
      DBLEF(RARG)  = RARG                                               SNGL0600
      KTRL    = KONTRL + 1                                              SNGL0820
      GO TO (2001,2002,2003),KTRL                                       SNGL0830
 2001 CONTINUE                                                          SNGL0930
      QABS    = ZER                                                     SNGL0980
      QRPR    = ZER                                                     SNGL0990
      COSTHE  = ZER                                                     SNGL1000
      ALBEDO  = ZER                                                     SNGL1010
      AREA    = PI*RADIUS**2                                            SNGL1020
      AREA1   = AREA/(FOU*PI)                                           SNGL1030
      ASSIGN 11 TO JUMP0                                                SNGL1040
      ASSIGN 17 TO JUMP1                                                SNGL1050
      ASSIGN 22 TO JUMP2                                                SNGL1060
      GO TO 10030                                                       SNGL1070
```

```
C      ENTRY      QMIESC(WAVE,RADIUS,INDEX,SIGMA,QEXT,QABS,QRPR,COSTHE,   SNGL1080
C     1ALBEDO)                                                           SNGL1090
 2002 CONTINUE                                                           SNGL1100
      AREA   = ONE                                                       SNGL1110
      AREA1  = ONE                                                       SNGL1120
      ASSIGN 10 TO JUMP0                                                 SNGL1130
      ASSIGN 16 TO JUMP1                                                 SNGL1140
      ASSIGN 21 TO JUMP2                                                 SNGL1150
      GO TO 10000                                                        SNGL1160
C     ENTRY      CMIESC(WAVE,RADIUS,INDEX,SIGMA,QEXT,QABS,QRPR,COSTHE,   SNGL1170
C     1ALBEDO)                                                           SNGL1180
 2003 CONTINUE                                                           SNGL1190
      AREA   = PI*RADIUS**2                                              SNGL1200
      AREA1  = AREA                                                      SNGL1210
      ASSIGN 10 TO JUMP0                                                 SNGL1220
      ASSIGN 16 TO JUMP1                                                 SNGL1230
      ASSIGN 21 TO JUMP2                                                 SNGL1240
      GO TO 10000                                                        SNGL1250
10000 CONTINUE                                                           SNGL1260
      JNDEX  = INDEX                                                     SNGL1270
      SIGR   = ZER                                                       SNGL1280
      SIGI   = ZER                                                       SNGL1290
      EXT    = ZER                                                       SNGL1300
      SCA    = ZER                                                       SNGL1310
      RPR    = ZER                                                       SNGL1320
      AR     = ZER                                                       SNGL1330
      AI     = ZER                                                       SNGL1340
      BR     = ZER                                                       SNGL1350
      BI     = ZER                                                       SNGL1360
      IF(LENGTH.EQ.0) LENGTH = NN                                        SNGL1370
      IF(LENGTH.EQ.NN) GO TO 11000                                       SNGL1380
      WRITE(6,10999)                                                     SNGL1390
10999 FORMAT(71H1ARRAYS IN CALLING PROGRAM DO NOT AGREE WITH THOSE IN SUSNGL1400
     1BROUTINE SPHERE)                                                   SNGL1410
      STOP                                                               SNGL1420
11000 CONTINUE                                                           SNGL1430
      X      = TWOPI*DBLEF(RADIUS)/DBLEF(WAVE)                           SNGL1440
      RX     = ONE/X                                                     SNGL1450
      RX2    = RX**2                                                     SNGL1460
      N      = 5 + IFIXF(X) + 5*IFIXF(EXPF(ALOGF(X/TWO)/THR))            SNGL1470
      N1     = N + 1                                                     SNGL1480
      M      = N - 1                                                     SNGL1490
      L      = N - 2                                                     SNGL1500
      IF(N.LE.NN) GO TO 1000                                             SNGL1510
      WRITE(6,999) WAVE,RADIUS,INDEX,X,N,LENGTH                          SNGL1520
  999 FORMAT(44H1SUBROUTINE SPHERE CALLED WITH WAVELENGTH = ,E15.8,      SNGL1530
     111H, RADIUS =,E15.8/24H INDEX OF REFRACTION = (,2E15.8,9H), KA = ,SNGL1540
     2E15.8,14H REQUIRES N = , I4, 32H ARRAYS ARE DIMENSIONED BY NN = ,  SNGL1550
     3I4/35H RECOMPILE WITH LARGER VALUE OF NN.)                         SNGL1560
      STOP                                                               SNGL1570
 1000 CONTINUE                                                           SNGL1580
      CHIZER = COSF(X)                                                   SNGL1590
      CHI(1) = SINF(X) + CHIZER*RX                                       SNGL1600
      CHI(2) =-CHIZER + CHI(1)*3.0*RX                                    SNGL1610
      CONST  = THR                                                       SNGL1620
      DO 1  I=3,N                                                        SNGL1630
```

```
            CONST   = CONST + TWO                                    SNGL1640
            CHI(I)  =-CHI(I-2) + CHI(I-1)*CONST*RX                   SNGL1650
          1 CONTINUE                                                 SNGL1660
            CONST   = FLOATF(N + LL + 2)                             SNGL1670
C-----DXR     = FLOAT(M + LL + 2)*RX - X/FLOAT(2*(N + LL) + 5)       SNGL1680
            DXR     = CONST*RX - X/(TWO*CONST + ONE)                 SNGL1690
C-----RATIO   = FLOAT(2*(N + LL) + 1)*RX                             SNGL1700
            RATIO   = (TWO*CONST - THR)*RX                           SNGL1710
C           DO 2  I=LL,1,-1                                          SNGL1720
            I       = LL + 1                                         SNGL1730
            DO 2 J = 1,LL                                            SNGL1740
```

```
C      ENTRY      QMIESC(WAVE,RADIUS,INDEX,SIGMA,QEXT,QABS,QRPR,COSTHE,     SNGL1080
C     1ALBEDO)                                                              SNGL1090
  2002 CONTINUE                                                             SNGL1100
       AREA    = ONE                                                        SNGL1110
       AREA1   = ONE                                                        SNGL1120
       ASSIGN 10 TO JUMP0                                                   SNGL1130
       ASSIGN 16 TO JUMP1                                                   SNGL1140
       ASSIGN 21 TO JUMP2                                                   SNGL1150
       GO TO 10000                                                          SNGL1160
C      ENTRY      CMIESC(WAVE,RADIUS,INDEX,SIGMA,QEXT,QABS,QRPR,COSTHE,     SNGL1170
C     1ALBEDO)                                                              SNGL1180
  2003 CONTINUE                                                             SNGL1190
       AREA    = PI*RADIUS**2                                               SNGL1200
       AREA1   = AREA                                                       SNGL1210
       ASSIGN 10 TO JUMP0                                                   SNGL1220
       ASSIGN 16 TO JUMP1                                                   SNGL1230
       ASSIGN 21 TO JUMP2                                                   SNGL1240
       GO TO 10000                                                          SNGL1250
 10000 CONTINUE                                                             SNGL1260
       JNDEX   = INDEX                                                      SNGL1270
       SIGR    = ZER                                                        SNGL1280
       SIGI    = ZER                                                        SNGL1290
       EXT     = ZER                                                        SNGL1300
       SCA     = ZER                                                        SNGL1310
       RPR     = ZER                                                        SNGL1320
       AR      = ZER                                                        SNGL1330
       AI      = ZER                                                        SNGL1340
       BR      = ZER                                                        SNGL1350
       BI      = ZER                                                        SNGL1360
       IF(LENGTH.EQ.0) LENGTH = NN                                          SNGL1370
       IF(LENGTH.EQ.NN) GO TO 11000                                         SNGL1380
       WRITE(6,10999)                                                       SNGL1390
 10999 FORMAT(71H1ARRAYS IN CALLING PROGRAM DO NOT AGREE WITH THOSE IN SUSNGL1400
      1BROUTINE SPHERE)                                                     SNGL1410
       STOP                                                                 SNGL1420
 11000 CONTINUE                                                             SNGL1430
       X       = TWOPI*DBLEF(RADIUS)/DBLEF(WAVE)                            SNGL1440
       RX      = ONE/X                                                      SNGL1450
       RX2     = RX**2                                                      SNGL1460
       N       = 5 + IFIXF(X) + 5*IFIXF(EXPF(ALOGF(X/TWO)/THR))             SNGL1470
       N1      = N + 1                                                      SNGL1480
       M       = N - 1                                                      SNGL1490
       L       = N - 2                                                      SNGL1500
       IF(N.LE.NN) GO TO 1000                                               SNGL1510
       WRITE(6,999) WAVE,RADIUS,INDEX,X,N,LENGTH                            SNGL1520
   999 FORMAT(44H1SUBROUTINE SPHERE CALLED WITH WAVELENGTH = ,E15.8,        SNGL1530
      111H, RADIUS =,E15.8/24H INDEX OF REFRACTION = (,2E15.8,9H), KA = ,SNGL1540
      2E15.8,14H REQUIRES N = , I4, 32H ARRAYS ARE DIMENSIONED BY NN = ,  SNGL1550
      3I4/35H RECOMPILE WITH LARGER VALUE OF NN.)                           SNGL1560
       STOP                                                                 SNGL1570
  1000 CONTINUE                                                             SNGL1580
       CHIZER = COSF(X)                                                     SNGL1590
       CHI(1) = SINF(X) + CHIZER*RX                                         SNGL1600
       CHI(2) =-CHIZER + CHI(1)*3.0*RX                                      SNGL1610
       CONST  = THR                                                         SNGL1620
       DO 1  I=3,N                                                          SNGL1630
```

```
      CONST  = CONST + TWO                                              SNGL1640
      CHI(I) =-CHI(I-2) + CHI(I-1)*CONST*RX                            SNGL1650
    1 CONTINUE                                                          SNGL1660
      CONST  = FLOATF(N + LL + 2)                                       SNGL1670
C-----DXR    = FLOAT(M + LL + 2)*RX - X/FLOAT(2*(N + LL) + 5)           SNGL1680
      DXR    = CONST*RX - X/(TWO*CONST + ONE)                           SNGL1690
C-----RATIO  = FLOAT(2*(N + LL) + 1)*RX                                 SNGL1700
      RATIO  = (TWO*CONST - THR)*RX                                     SNGL1710
C     DO 2  I=LL,1,-1                                                   SNGL1720
      I      = LL + 1                                                   SNGL1730
      DO 2 J = 1,LL                                                     SNGL1740
      I      = I - 1                                                    SNGL1750
      CONST  = CONST - ONE                                              SNGL1760
C-----CONST  = FLOAT(I + N + 1)                                         SNGL1770
      FAC    = CONST*RX                                                 SNGL1780
      DXR    = FAC - ONE/(DXR + FAC)                                    SNGL1790
C-----RATIO  = FLOAT(2*(N + I) - 1)*RX - ONE/RATIO                      SNGL1800
      RATIO  =(TWO*CONST - THR)*RX - ONE/RATIO                          SNGL1810
    2 CONTINUE                                                          SNGL1820
      PSI(N) = SMALL                                                    SNGL1830
      PSI(M) = SMALL*RATIO                                              SNGL1840
C-----CONST  = FLOAT(2*(N + LL) + 1)                                    SNGL1850
      CONST  = TWO*CONST - THR                                          SNGL1860
C-----CONST  = FLOAT(2*N + 1)                                           SNGL1870
C     DO 3  I=L,1,-1                                                    SNGL1880
      I      = L + 1                                                    SNGL1890
      DO 3 J = 1,L                                                      SNGL1900
      I      = I - 1                                                    SNGL1910
C-----CONST  = FLOAT(2*I + 3)                                           SNGL1920
      CONST  = CONST - TWO                                              SNGL1930
      PSI(I) =-PSI(I+2) + PSI(I+1)*CONST*RX                            SNGL1940
    3 CONTINUE                                                          SNGL1950
      XL     = ABSF(PSI(1))                                             SNGL1960
      IF(ABSF(PSI(2)).GT.ABSF(PSI(1))) XL = ABSF(PSI(2))               SNGL1970
      A1     = PSI(1)/XL                                                SNGL1980
      A2     = PSI(2)/XL                                                SNGL1990
      FACTOR = ONE/(XL*SQRTF((THR*RX*A1 - A2)**2                        SNGL2000
     1           + ((THR*RX**2 - ONE)*A1 - A2*RX)**2))                  SNGL2010
      DO 4  I=1,N                                                       SNGL2020
      PSI(I) = FACTOR*PSI(I)                                            SNGL2030
      HOLD   = PSI(I)                                                   SNGL2040
      FAC    = PSI(I)/(PSI(I)**2 + CHI(I)**2)                           SNGL2050
      PSI(I) = FAC*PSI(I)                                               SNGL2060
      CHI(I) = FAC*CHI(I)                                               SNGL2070
    4 CONTINUE                                                          SNGL2080
      FAC    = ONE/(ONE + RX**2)                                        SNGL2090
      GR(1)  =-RX + FAC*RX                                              SNGL2100
      GI(1)  = FAC                                                      SNGL2110
      CONST  = ONE                                                      SNGL2120
      DO 5  I=2,N                                                       SNGL2130
C-----CONST  = FLOAT(I)                                                 SNGL2140
      CONST  = CONST + ONE                                              SNGL2150
      RXN    = CONST*RX                                                 SNGL2160
      DEN    = RXN - GR(I-1)                                            SNGL2170
      FAC    = ONE/(DEN**2 + GI(I-1)**2)                                SNGL2180
      GR(I)  =-RXN + FAC*DEN                                            SNGL2190
```

```
      GI(I)   = FAC*GI(I-1)                                            SNGL2200
    5 CONTINUE                                                         SNGL2210
      FR      = REALF(INDEX)                                           SNGL2220
      FI      = ABSF(AIMAF(INDEX))                                     SNGL2230
      FAC     = FR**2 + FI**2                                          SNGL2240
      IF(FAC.GT.D10P6) GO TO 15                                        SNGL2250
      ENR     = FR/FAC                                                 SNGL2260
      ENI     =-FI/FAC                                                 SNGL2270
      Z       = X*SQRTF(FAC)                                           SNGL2280
      YR      = X*FR                                                   SNGL2290
      YI      = X*FI                                                   SNGL2300
      FAC     = ONE/(YR**2 + YI**2)                                    SNGL2310
      RYR     = FAC*YR                                                 SNGL2320
      RYI     =-FAC*YI                                                 SNGL2330
      DYR     = ZER                                                    SNGL2340
      DYI     = ZER                                                    SNGL2350
      IF(Z.GT.GREAT) GO TO 60                                          SNGL2360
      K       = 5 + IFIXF(Z) + 5*IFIXF(EXPF(ALOGF(Z/TWO)/THR))         SNGL2370
      IF(2*K.LT.5*N) GO TO 7                                           SNGL2380
      FAC     = TWO*YI                                                 SNGL2390
      IF(FAC.GT.FIFTY) GO TO 60                                        SNGL2400
      FAC     = EXPF(FAC)                                              SNGL2410
      CI      =-HALF*FAC + HALF/FAC                                    SNGL2420
      DR      = HALF*FAC + HALF/FAC                                    SNGL2430
      FAC     = TWO*YR                                                 SNGL2440
      CR      = SINF(FAC)                                              SNGL2450
      FAC     = ONE/(DR - COSF(FAC))                                   SNGL2460
      DYR     = FAC*CR                                                 SNGL2470
      DYI     = FAC*CI                                                 SNGL2480
   60 CONTINUE                                                         SNGL2490
      CONST   = ZER                                                    SNGL2500
      DO 6  I=1,NI                                                     SNGL2510
C-----CONST   = FLOAT(I)                                               SNGL2520
      CONST   = CONST + ONE                                            SNGL2530
      FR      = CONST*RYR                                              SNGL2540
      FI      = CONST*RYI                                              SNGL2550
      DR      = FR - DYR                                               SNGL2560
      DI      = FI - DYI                                               SNGL2570
      FAC     = ONE/(DR**2 + DI**2)                                    SNGL2580
      DYR     =-FR + FAC*DR                                            SNGL2590
      DYI     =-FI - FAC*DI                                            SNGL2600
    6 CONTINUL                                                         SNGL2610
      GO TO 9                                                          SNGL2620
    7 CONTINUE                                                         SNGL2630
      J       = MAX0(K,N)+ 5                                           SNGL2640
      CONST   = FLOAT(J+1)                                             SNGL2650
      FR      = CONST*RYR                                              SNGL2660
      FI      = CONST*RYI                                              SNGL2670
C-----FAC     = ONE/FLOAT(2*J + 3)                                     SNGL2680
      FAC     = ONE/(TWO*CONST + ONE)                                  SNGL2690
      DYR     = FR - FAC*YR                                            SNGL2700
      DYI     = FI + FAC*YI                                            SNGL2710
C     DO 8  I=J,NI,-1                                                  SNGL2720
C     I       = J + 1                                                  SNGL2730
      DO 8 II=NI,J                                                     SNGL2740
C     I       = I - 1                                                  SNGL2760
```

```
C-----CONST  = FLOAT(I+1)                                    SNGL2760
      CONST  = CONST - ONE                                   SNGL2770
      FR     = CONST*RYR                                     SNGL2780
      FI     = CONST*RYI                                     SNGL2790
      DR     = DYR + FR                                      SNGL2800
      DI     = DYI + FI                                      SNGL2810
      FAC    = ONE/(DR**2 + DI**2)                           SNGL2820
      DYR    = FR - FAC*DR                                   SNGL2830
      DYI    = FI + FAC*DI                                   SNGL2840
    8 CONTINUE                                               SNGL2850
      GO TO 9                                                SNGL2860
    9 CONTINUE                                               SNGL2870
      CONST  = FLOATF(N + 2)                                 SNGL2880
C     DO 11 I=N,1,-1                                         SNGL2890
      I      = N + 1                                         SNGL2900
      DO 11 J = 1,N                                          SNGL2910
      I      = I - 1                                         SNGL2920
      TIGN   =-TIGN                                          SNGL2930
      ARS    = AR                                            SNGL2940
      AIS    = AI                                            SNGL2950
      BRS    = BR                                            SNGL2960
      BIS    = BI                                            SNGL2970
C-----CONST  = FLOAT(I + 1)                                  SNGL2980
      CONST  = CONST - ONE                                   SNGL2990
      FAC    = CONST*RX                                      SNGL3000
      DXR    = FAC - ONE/(DXR + FAC)                         SNGL3010
      FR     = CONST*RYR                                     SNGL3020
      FI     = CONST*RYI                                     SNGL3030
      DR     = DYR + FR                                      SNGL3040
      DI     = DYI + FI                                      SNGL3050
      FAC    = ONE/(DR**2 + DI**2)                           SNGL3060
      DYR    = FR - FAC*DR                                   SNGL3070
      DYI    = FI + FAC*DI                                   SNGL3080
      DR     = ENR*DYR - ENI*DYI                             SNGL3090
      DI     = ENR*DYI + ENI*DYR                             SNGL3100
      CR     = DXR - DR                                      SNGL3110
      CI     =     - DI                                      SNGL3120
      DR     = GR(I) - DR                                    SNGL3130
      DI     = GI(I) - DI                                    SNGL3140
      FAC    = ONE/(DR**2 + DI**2)                           SNGL3150
      TR     = FAC*(CR*DR + CI*DI)                           SNGL3160
      TI     = FAC*(CI*DR - CR*DI)                           SNGL3170
      AR     = TR*PSI(I) - TI*CHI(I)                         SNGL3180
      AI     = TR*CHI(I) + TI*PSI(I)                         SNGL3190
      CR     = DYR - ENR*DXR                                 SNGL3200
      CI     = DYI - ENI*DXR                                 SNGL3210
      DR     = DYR - (ENR*GR(I) - ENI*GI(I))                 SNGL3220
      DI     = DYI - (ENR*GI(I) + ENI*GR(I))                 SNGL3230
      FAC    = ONE/(DR**2 + DI**2)                           SNGL3240
      TR     = FAC*(CR*DR + CI*DI)                           SNGL3250
      TI     = FAC*(CI*DR - CR*DI)                           SNGL3260
      BR     = TR*PSI(I) - TI*CHI(I)                         SNGL3270
      BI     = TR*CHI(I) + TI*PSI(I)                         SNGL3280
      A(I,1) = AR                                            SNGL3290
      A(I,2) = AI                                            SNGL3300
      B(I,1) = BR                                            SNGL3310
```

```
      B(I,2) = BI                                                       SNGL3320
C-----FAC    = FLOAT(2*I + 1)                                           SNGL3330
      FAC    = TWO*CONST - ONE                                          SNGL3340
      SIGR   = SIGR + TIGN*FAC*(AR - BR)                                SNGL3350
      SIGI   = SIGI + TIGN*FAC*(AI - BI)                                SNGL3360
      EXT    = EXT + FAC*(AR + BR)                                      SNGL3370
      GO TO JUMPO,(10,11)                                               SNGL3380
   10 SCA    = SCA + FAC*(AR**2 + AI**2 + BR**2 + BI**2)                SNGL3390
      FAC    = FLOAT(I*(I+2))/FLOAT(I+1)                                SNGL3400
C-----AI     = FLOAT(I*(I+2))/FLOAT(I+1)                                SNGL3410
      AI     =(TWO*CONST - ONE)/(CONST - ONE)                           SNGL3420
C-----A2     = FLOAT(2*I+1)/FLOAT(I*(I+1))                              SNGL3430
      A2     =(CONST - ONE)*(CONST + ONE)/CONST                         SNGL3440
      RPR    = RPR + AI*(AR*ARS + AI*AIS + BR*BRS + BI*BIS) +           SNGL3450
     1             A2*(AR*BR*AI*BI)                                     SNGL3460
   11 CONTINUE                                                          SNGL3470
      GO TO 20                                                          SNGL3480
   15 CONTINUE                                                          SNGL3490
      DYR    = ZER                                                      SNGL3500
      DYI    = ZER                                                      SNGL3510
      CONST  = FLOATF(N + 2)                                            SNGL3520
C     DO 17 I=N,1,-1                                                    SNGL3530
      I      = N + 1                                                    SNGL3540
      DO 17 J = 1,N                                                     SNGL3550
      I      = I - 1                                                    SNGL3560
      TIGN   =-TIGN                                                     SNGL3570
      ARS    = AR                                                       SNGL3580
      AIS    = AI                                                       SNGL3590
      BRS    = BR                                                       SNGL3600
      BIS    = BI                                                       SNGL3610
C-----CONST  = FLOAT(I + 1)                                             SNGL3620
      CONST  = CONST - ONE                                              SNGL3630
      FAC    = CONST*RX                                                 SNGL3640
      DXR    = FAC - ONE/(DXR + FAC)                                    SNGL3650
      FAC    = DXR/(GR(I)**2 + GI(I)**2)                                SNGL3660
      TR     = FAC*GR(I)                                                SNGL3670
      TI     =-FAC*GI(I)                                                SNGL3680
      AR     = TR*PSI(I) - TI*CHI(I)                                    SNGL3690
      AI     = TR*CHI(I) + TI*PSI(I)                                    SNGL3700
      BR     = PSI(I)                                                   SNGL3710
      BI     = CHI(I)                                                   SNGL3720
      A(I,1) = AR                                                       SNGL3730
      A(I,2) = AI                                                       SNGL3740
      B(I,1) = BR                                                       SNGL3750
      B(I,2) = BI                                                       SNGL3760
C-----FAC    = FLOAT(2*I + 1)                                           SNGL3770
      FAC    = TWO*CONST - ONE                                          SNGL3780
      SIGR   = SIGR + TIGN*FAC*(AR - BR)                                SNGL3790
      SIGI   = SIGI + TIGN*FAC*(AI - BI)                                SNGL3800
      EXT    = EXT + FAC*(AR + BR)                                      SNGL3810
      GO TO JUMP1,(16,17)                                               SNGL3820
   16 SCA    = SCA + FAC*(AR**2 + AI**2 + BR**2 + BI**2)                SNGL3830
C-----AI     = FLOAT(I*(I+2))/FLOAT(I+1)                                SNGL3840
      AI     =(TWO*CONST - ONE)/(CONST - ONE)                           SNGL3850
C-----A2     = FLOAT(2*I+1)/FLOAT(I*(I+1))                              SNGL3860
      A2     =(CONST - ONE)*(CONST + ONE)/CONST                         SNGL3870
```

```
          RPR     = RPR + AI*(AR*ARS + AI*AIS + BR*BRS + BI*BIS) +              SNGL3880
         1                AR*(AR*BR*AI*BI)                                      SNGL3890
   17 CONTINUE                                                                  SNGL3900
      GO TO 20                                                                  SNGL3910
   20 CONTINUE                                                                  SNGL3920
      SIGR    =-TIGN*SIGR                                                       SNGL3930
      SIGI    =-TIGN*SIGI                                                       SNGL3940
      SIGMA   = SNGLF((SIGR**2 + SIGI**2)*RX2*AREA))                            SNGL3950
      CEXT    = TWO*EXT*RX2*AREA                                                SNGL3960
      QEXT    = SNGLF(CEXT)                                                     SNGL3970
      XKA     = SNGLF(X)                                                        SNGL397A
      GO TO JUMP2,(21,22)                                                       SNGL3980
   21 CSCA    = TWO*SCA*RX2*AREA                                                SNGL3990
      QSCA    = SNGLF(CSCA)                                                     SNGL4000
      CCAB    = CEXT - CSCA                                                     SNGL4010
      QABS    = SNGLF(CCAB)                                                     SNGL4020
      CRPR    = CEXT - FOU*RPR*RX2*AREA                                         SNGL4030
      QRPR    = SNGLF(CRPR)                                                     SNGL4040
      COSTHE  = (QEXT - QRPR)/(QEXT - QABS)                                     SNGL4050
      ALBEDO  = 1.0/(1.0 + QABS/QSCA)                                           SNGL4060
   22 RETURN                                                                    SNGL4070
      END                                                                       SNGL4080
```

PRT,S TARGET

```
LEE-ALBIN202*TPFS.TARGET
              SUBROUTINE TARGET
              INCLUDE GEOM,LIST
              REAL I11,I12,I13,I21,I22,I23,I31,I32,I33,I,J,K,L,N1,N2,NUMBER,
             1 LOWLIM,M,N
              DIMENSION PDUM(3),XDUM(3),XTDUM(4)
              RANGE(X1,Y1,Z1,X2,Y2,Z2)=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
              ATRIG(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3)=
             *  SQRT(  (X1*(Y2-Y3)+X2*(Y3-Y1)+X3*(Y1-Y2))**2
             *          +(Z1*(X2-X3)+Z2*(X3-X1)+Z3*(X1-X2))**2
             *          +(Y1*(Z2-Z3)+Y2*(Z3-Z1)+Y3*(Z1-Z2))**2  )/2.0
              MTX(U11,U12,U13,U21,U22,U23,U31,U32,U33) =
             *     U11*U12*U13 + U13*U21*U32 + U31*U12*U23
             *   -U13*U22*U31 - U32*U23*U11 - U33*U12*U21
              PI=3.14159265
              HALFPI=PI/2.0
              TWOPI=PI*2.0
              NDATA=0
C*****************************************************
C
C            DETERMINATION OF COEFFICIENTS FOR QUADRIC
C        EQUATIONS DESCRIBING THREE DIMENSIONAL SURFACES,
C            THEIR CONSTRAINT PLANES,SKEWED PLANES,AND
C          DETERMINATION OF VIEW POINT VECTOR COMPONENTS
C        AND VIEW POINT AREAS FOR THE VARIOUS GEOMETRIES
C
C*** NTMAX= MAX. NUMBER OF TARGETS ALLOWED IN THIS PROGRAM (.GE. NTARGT)
C*** NRING=NUMBER OF RINGS ON THE CONSTRAINT ANNULAR DISK
C             (.GE. MAX(NRING1,NRING2))
C*** NSIDEA= TOTAL NUMBER OF VIEW POINT AREAS ON THE SIDE SURFACE OF
C               TARGET  (.GE.(NN1*NN2) )
C*** NTARGT=NUMBER OF TARGETS
C
C*****
C
C       CYLINDER
C
C          IPTION=1
C          NN1=NUMBER OF VIEW POINTS ALONG ARC LENGTH
C          NN2=NUMBER OF VIEW POINTS ALONG AXIAL DIRECTION
C          NRING1=NUMBER OF VIEW POINT AREAS ALONG RADIAL DIRECTION ON THE
C                CONSTRAINT ANNULAR DISK PASSING THRU POINT P1(X10,X20,X30)
C          NRING2=NUMBER OF VIEW POINT AREAS ALONG RADIAL DIRECTION ON THE
C                CONSTRAINT ANNULAR DISK PASSING THRU POINT P2(X40,X50,X60)
C          RADISK=RADIUS OF THE CYLINDER
C          RCP1=RADIUS OF THE INNER CIRCLE ON THE CONTRAINT ANNULAR DISK
C                PASSING THROUGH POINT P1(X10,X20,X30).
C          RCP2=RADIUS OF THE INNER CIRCLE ON THE CONTRAINT ANNULAR DISK
C                PASSING THROUGH POINT P2(X40,X50,X60).
C          X10,X20,X30 ARE THE COORDINATES OF THE CENTER OF THE TOP
C          X40,X50,X60 ARE THE COORDINATES OF THE CENTER OF THE BASE
C          X70,X80,X90 ARE THE COORDINATES OF AN ARBITRARY POINT NOT
C                       ON THE AXIS OF THE CYLINDER
C
C       FRUSTUM
C
```

```
C         IPTION=2
C         NN1=NUMBER OF VIEW POINTS ALONG ARC LENGTH
C         NN2=NUMBER OF VIEW POINTS ALONG AXIAL DIRECTION
C         NRING1=OPTIONAL, DOES NOT APPLY
C         NRING2=NUMBER OF VIEW POINT AREAS ALONG RADIAL DIRECTION ON THE
C               CONSTRAINT ANNULAR DISK PASSING THRU POINT P2(X40,X50,X60)
C         RADISK=RADIUS OF THE CONSTRAINT DISK PASSING THRU P2
C         RCP1=OPTIONAL, DOES NOT APPLY
C         RCP2=RADIUS OF THE INNER CIRCLE ON THE CONTRAINT ANNULAR DISK
C               PASSING THROUGH POINT P2(X40,X50,X60).
C         RADISU=RADIUS OF THE CONSTRAINT DISK THRU POINT P1(X10,X20,X30)
C         X10,X20,X30 ARE THE COORDINATES OF THE CENTER OF THE TOP
C         X40,X50,X60 ARE THE COORDINATES OF THE CENTER OF THE BASE
C         X70,X80,X90 ARE THE COORDINATES OF THE ARBITRARY POINT NOT
C                     ON THE AXIS OF THE FRUSTUM
C
C     PARABOLOID
C
C         IPTION=3
C         NN1=NUMBER OF VIEW POINTS ALONG ARC LENGTH
C         NN2=NUMBER OF VIEW POINTS ALONG AXIAL DIRECTION
C         NRING1=OPTIONAL, DOES NOT APPLY
C         NRING2=NUMBER OF VIEW POINT AREAS ALONG RADIAL DIRECTION ON THE
C               CONSTRAINT ANNULAR DISK PASSING THRU POINT P2(X40,X50,X60)
C         RADISK=RADIUS OF THE CONSTRAINT DISK PASSING THRU P2
C         RCP1=OPTIONAL, DOES NOT APPLY
C         RCP2=RADIUS OF THE INNER CIRCLE ON THE CONTRAINT ANNULAR DISK
C               PASSING THROUGH POINT P2(X40,X50,X60).
C         X10,X20,X30 ARE THE COORDINATES OF THE VERTEX
C         X40,X50,X60 ARE THE COORDINATES OF THE CENTER OF THE BASE
C         X70,X80,X90 ARE THE COORDINATES OF THE ARBITRARY POINT NOT
C                     ON THE AXIS OF THE PARABOLOID
C
C     ELLIPSOID   (SPHERE IS A SPECIAL CASE)
C
C         IPTION=4
C         NN1=NUMBER OF VIEW POINTS ALONG ARC LENGTH
C         NN2=NUMBER OF VIEW POINTS ALONG P1-P2 AXIAL DIRECTION
C         NRING1,NRING2=OPTIONAL, DOES NOT APPLY
C         RAD=NUMERICAL VALUE OPTIONAL,DOES NOT APPLY
C         RCP1,RCP2=OPTIONAL, DOES NOT APPLY
C         X10,X20,X30 ARE THE COORDINATES OF THE CENTER
C         X40,X50,X60 ARE THE COORDINATES OF THE POLE
C         X70,X80,X90 ARE THE COORDINATES OF THE ZERO MERIDIAN
C
C     PLANE
C
C         IPTION=5
C         NN1=NUMBER OF VIEW POINTS ALONG P1-P2 LINE
C         NN2=NUMBER OF VIEW POINTS ALONG P2-P3 LINE
C         NRING1,NRING2=OPTIONAL, DOES NOT APPLY
C         RAD=NUMERICAL VALUE OPTIONAL,DOES NOT APPLY
C         RCP1,RCP2=OPTIONAL, DOES NOT APPLY
C               THE FOLLOWING ARE P1,P2,P3 RESPECTIVELY
C         X10,X20,X30 ARE THE COORDINATES OF A CORNER OF THE PLANE
C         X40,X50,X60 ARE THE COORDINATES OF THE NEXT CORNER.
```

```
C                          CLOCKWISE ABOUT THE PLANE, LOOKING IN THE DIRECTION
C                          OF THE NEGATIVE NORMAL
C             X70,X80,X90 ARE THE COORDINATES OF THE THIRD CORNER.
C                          CLOCKWISE ABOUT THE PLANE, LOOKING IN THE DIRECTION
C                          OF THE NEGATIVE NORMAL
C
C     ANNULAR DISK
C
C          IPTION=6
C          NN1=NUMBER OF VIEW POINTS IN ARC DIRECTION
C          NN2=NUMBER OF VIEW POINTS IN RADIAL DIRECTION
C          NRING1,NRING2=OPTIONAL, DOES NOT APPLY
C          RADISK=OUTER RADIUS OF THE DISK
C          RCP1=INNER RADIUS OF THE ANNULAR DISK
C          RCP2=OPTIONAL, DOES NOT APPLY
C          X10,X20,X30 ARE THE COORDINATES OF P1, CENTER OF THE DISK
C          X40,X50,X60 IS A POINT ON THE DISK, P2. VIEW POINTS ARE COUNTED
C                          FROM P1-P2 LINE IN RIGHT-HAND-RULE SENSE WITH ITS NORMAL
C          X70,X80,X90 IS THE DIRECTION COSINE OF THE POSITIVE NORMAL
C
C**************************************************************
   10 CONTINUE
      NDATA=NDATA+1
      READ (5,251) IPTION,NN1,NN2,NRING1,NRING2,RADISK,RCP1,RCP2,RADISU
      IF (IPTION.LE.0)                                   RETURN
      READ (5,252) X10,X20,X30,X40,X50,X60,X70,X80,X90
      WRITE (6,201) NDATA
      WRITE (6,251) IPTION,NN1,NN2,NRING1,NRING2,RADISK,RCP1,RCP2,RADISU
      WRITE (6,252) X10,X20,X30,X40,X50,X60,X70,X80,X90
      NTARGT=NDATA
      IF (NN1.LE.1) NN1=1
      IF (NN2.LE.1) NN2=1
      IF (NRING1.LE.1) NRING1=1
      IF (NRING2.LE.1) NRING2=1
      NAREA(NDATA,1)=NN1*NN2
      NAREA(NDATA,2)=NN1*NRING1
      NAREA(NDATA,3)=NN1*NRING2
      RADISK=ABS(RADISK)
      RCP1=ABS(RCP1)
      RCP2=ABS(RCP2)
      IF (IPTION.EQ.4) GO TO 901
      N1=1.0/FLOAT(NN1)
      N2=1.0/FLOAT(NN2)
      GO TO 902
  901 N1=1.0/FLOAT(NN2)
      N2=1.0/FLOAT(NN1)
  902 L=SQRT((X40-X10)**2+(X50-X20)**2+(X60-X30)**2)
      IF (IPTION.EQ.1 .OR. IPTION.EQ.6)                GO TO 801
      IF (IPTION.EQ.2) GO TO 802
      IF (IPTION.EQ.3) GO TO 803
      GO TO 903
  801 RAD=RADISK
      GO TO 903
  802 RAD=(RADISK-RADISU)/L
      RBONDU(NDATA)=RADISU
      GO TO 903
```

```
      803 RAD=(RADISK**2)/L
      903 RBOND(NDATA)=RADISK
          IBODY(NDATA)=IPTION
          DATA(NDATA,1)=NN1
          DATA(NDATA,2)=NN2
          DATA(NDATA,3)=L
          DATA(NDATA,4)=RAD
          DATA(NDATA,5)=X10
          DATA(NDATA,6)=X20
          DATA(NDATA,7)=X30
          DATA(NDATA,8)=X40
          DATA(NDATA,9)=X50
          DATA(NDATA,10)=X60
          DATA(NDATA,11)=X70
          DATA(NDATA,12)=X80
          DATA(NDATA,13)=X90
          RRING(NDATA,1)=RCP1
          RRING(NDATA,2)=RCP2
          IF (IPTION.NE.2)                          GO TO 905
          CONEHT=L
          CV1=(X10-X40)/CONEHT
          CV2=(X20-X50)/CONEHT
          CV3=(X30-X60)/CONEHT
          ADDHT=CONEHT*RADISU/(RADISK-RADISU)
          X10=X10+CV1*ADDHT
          X20=X20+CV2*ADDHT
          X30=X30*CV3*ADDHT
      905 CONTINUE
          B1=X40-X10
          B2=X50-X20
          B3=X60-X30
          E1=X70-X40
          E2=X80-X50
          E3=X90-X60
          R1=X70-X10
          R2=X80-X20
          R3=X90-X30
          BMAG=SQRT(B1**2+B2**2+B3**2)
          EMAG=SQRT(E1**2+E2**2+E3**2)
          RMAG=SQRT(R1**2+R2**2+R3**2)
          IF(IPTION-5)  101,111,100
      101 BETA=ACOS(B1/((B1**2+B3**2)**.5))
          IF(B3)103,103,102
      102 BETA=-BETA
      103 THETA=HALFPI-ACOS(B2/BMAG)
          I11=COS(THETA)*COS(BETA)
          I12=SIN(THETA)
          I13=-SIN(BETA)*COS(THETA)
          I21=-SIN(THETA)*COS(BETA)
          I22=COS(THETA)
          I23=SIN(THETA)*SIN(BETA)
          I31=SIN(BETA)
          I32=0.0
          I33=COS(BETA)
          WRITE(6,222)
          WRITE(6,223) I11,I12,I13
```

```
      WRITE(6,203) I21,I22,I23
      WRITE(6,203) I31,I32,I33
      I=I11*X10+I12*X20+I13*X30
      J=I21*X10+I22*X20+I23*X30
      K=I31*X10+I32*X20+I33*X30
      IF(IPTION-4) 104,110,104
  104 IF(IPTION-3) 105,108,105
  105 IF(IPTION-2) 106,107,106
  106 C1=I21**2+I31**2
      C2=I22**2+I32**2
      C3=I23**2+I33**2
      C4=2.0*(I21*I22+I31*I32)
      C5=2.0*(I21*I23+I31*I33)
      C6=2.0*(I22*I23+I32*I33)
      C7=-2.0*(J*I21+K*I31)
      C8=-2.0*(J*I22+K*I32)
      C9=-2.0*(J*I23+K*I33)
      CONST=J**2+K**2-RAD**2
      WRITE(6,204)
      GO TO 109
  107 C1=-(RAD*I11)**2+I21**2+I31**2
      C2=-(RAD*I12)**2+I22**2+I32**2
      C3=-(RAD*I13)**2+I23**2+I33**2
      C4=2.0*(-RAD**2*I11*I12+I21*I22+I31*I32)
      C5=2.0*(-RAD**2*I11*I13+I21*I23+I31*I33)
      C6=2.0*(-RAD**2*I12*I13+I22*I23+I32*I33)
      C7=-2.0*(-I*RAD**2*I11+J*I21+K*I31)
      C8=-2.0*(-I*RAD**2*I12+J*I22+K*I32)
      C9=-2.0*(-I*RAD**2*I13+J*I23+K*I33)
      CONST=-(I*RAD)**2+J**2+K**2
      WRITE(6,205)
      GO TO 109
  108 C1=I21**2+I31**2
      C2=I22**2+I32**2
      C3=I23**2+I33**2
      C4=2.0*(I21*I22+I31*I32)
      C5=2.0*(I21*I23+I31*I33)
      C6=2.0*(I22*I23+I32*I33)
      RAD2=RAD/2.0
      C7=-2.0*(J*I21+K*I31+RAD2*I11)
      C8=-2.0*(J*I22+K*I32+RAD2*I12)
      C9=-2.0*(J*I23+K*I33+RAD2*I13)
      CONST=J**2+K**2+2.0*RAD2*I
      WRITE(6,206)
  109 WRITE(6,607) C1,C2,C3,C4,C5,C6,C7,C8,C9,CONST,RBOND(NDATA)
     *          ,RBONDU(NDATA)
      X01=L*B1/BMAG
      X02=L*B2/BMAG
      X03=L*B3/BMAG
      X14=X40-X01
      X25=X50-X02
      X36=X60-X03
      C11=I11
      C12=I12
      C13=I13
      CBASE=I11*X40+I12*X50+I13*X60
```

```
      CEND=I11*X14+I12*X25+I13*X36
      WRITE(6,208)
      WRITE(6,209) C11, C12, C13, CEND,CBASE
      DISKEQ(NDATA,1)=C11
      DISKEQ(NDATA,2)=C12
      DISKEQ(NDATA,3)=C13
      DISKEQ(NDATA,4)=-CEND
      DISKEQ(NDATA,5)=-CBASE
      NCRING(NDATA,1)=NRING1
      NCRING(NDATA,2)=NRING2
      DRANG=2.0*PI*N1
      DPHIC(NDATA)=DRANG
      RRCP1=RBOND(NDATA)-RCP1
      IF (IPTION.EQ.2)        RRCP1=RBONDU(NDATA)-RCP1
      RRCP2=RBOND(NDATA)-RCP2
      IF (NRING1.LE.0 .OR. IPTION.GT.2)          GO TO 922
      DR1=RRCP1/FLOAT(NRING1)
      GO TO 925
  922 DR1=0.0
  925 IF (NRING2.LE.0) GO TO 923
      DR2=RRCP2/FLOAT(NRING2)
      GO TO 924
  923 DR2=0.0
  924 DO 930 JJ=1,2
      IF (JJ.EQ.1 .AND. IPTION.GT.2)           GO TO 930
      IF (JJ.EQ.2) GO TO 931
      DR=DR1
      RMEAN=RCP1+DR/2.0
      NRG=NRING1
      GO TO 932
  931 DR=DR2
      RMEAN=RCP2+DR/2.0
      NRG=NRING2
  932 DO 920 KK=1,NRG
      CRING(NDATA,KK,JJ)=RMEAN*DRANG*DR
      RMEANC(NDATA,KK,JJ)=RMEAN
      RMEAN=RMEAN+DR
  920 CONTINUE
  930 CONTINUE
      U12X=B1/BMAG
      U12Y=B2/BMAG
      U12Z=B3/BMAG
      P12U12=R1*U12X+R2*U12Y+R3*U12Z
      P1PAX=P12U12*U12X
      P1PAY=P12U12*U12Y
      P1PAZ=P12U12*U12Z
      P13MAX=R1-P1PAX
      P13MAY=R2-P1PAY
      P13MAZ=R3-P1PAZ
      P13MA=RANGE(P13MAX,P13MAY,P13MAZ,0.0,0.0,0.0)
      UA3X=P13MAX/P13MA
      UA3Y=P13MAY/P13MA
      UA3Z=P13MAZ/P13MA
      U25X=UA3Y*U12Z-UA3Z*U12Y
      U25Y=UA3Z*U12X-UA3X*U12Z
      U25Z=UA3X*U12Y-UA3Y*U12X
```

```
              TREF(NDATA,1,2)=RBOND(NDATA)*UA3X
              TREF(NDATA,2,2)=RBOND(NDATA)*UA3Y
              TREF(NDATA,3,2)=RBOND(NDATA)*UA3Z
              SREF(NDATA,1,2)=RBOND(NDATA)*U2SX
              SREF(NDATA,2,2)=RBOND(NDATA)*U2SY
              SREF(NDATA,3,2)=RBOND(NDATA)*U2SZ
              TTX=TREF(NDATA,1,2)+X40
              TTY=TREF(NDATA,2,2)+X50
              TTZ=TREF(NDATA,3,2)+X60
              SSX=SREF(NDATA,1,2)+X40
              SSY=SREF(NDATA,2,2)+X50
              SSZ=SREF(NDATA,3,2)+X60
              WRITE(6,941) TTX,TTY,TTZ,SSX,SSY,SSZ
              IF (IPTION.GT.2)           GO TO 921
              IF (IPTION.EQ.2)           RSAVE=RBOND(NDATA)
              IF (IPTION.EQ.2)           RBOND(NDATA)=RBONDU(NDATA)
              TREF(NDATA,1,1)=RBOND(NDATA)*UA3X
              TREF(NDATA,2,1)=RBOND(NDATA)*UA3Y
              TREF(NDATA,3,1)=RBOND(NDATA)*UA3Z
              SREF(NDATA,1,1)=RBOND(NDATA)*U2SX
              SREF(NDATA,2,1)=RBOND(NDATA)*U2SY
              SREF(NDATA,3,1)=RBOND(NDATA)*U2SZ
              IF (IPTION.EQ.2)           RBOND(NDATA)=RSAVE
          921 CONTINUE
              WRITE(6,333)
              GO TO 112
          110 C1=(RMAG*I11)**2+BMAG**2*(I21**2+I31**2)
              C2=(RMAG*I12)**2+BMAG**2*(I22**2+I32**2)
              C3=(RMAG*I13)**2+BMAG**2*(I23**2+I33**2)
              C4=2.0*(RMAG**2*I11*I12+BMAG**2*(I21*I22+I31*I32)   )
              C5=2.0*(RMAG**2*I11*I13+BMAG**2*(I21*I23+I31*I33)   )
              C6=2.0*(RMAG**2*I12*I13+BMAG**2*(I22*I23+I32*I33)   ).
              C7=-2.0*(RMAG**2*I*I11+BMAG**2*(J*I21+K*I31))
              C8=-2.0*(RMAG**2*I*I12+BMAG**2*(J*I22+K*I32))
              C9=-2.0*(RMAG**2*I*I13+BMAG**2*(J*I23+K*I33))
              CONST=(RMAG*I)**2+BMAG**2*(J**2+K**2)-(RMAG*BMAG)**2
              WRITE(6,210)
              WRITE(6,207) C1,C2,C3,C4,C5,C6,C7,C8,C9,CONST
              WRITE(6,333)
              GO TO 112
          111 C1=X20*E3+X80*B3-X50*R3
              C2=X30*E1+X90*B1-X60*R1
              C3=X10*E2 +X70*B2-X40*R2
              CONST=X10*(X60*X80-X50*X90)+X20*(X40*X90-X60*X70)+X30*(X50*X70-X40
             1*X80)
              WRITE(6,211)
              X1A=X10+X70-X40
              X2A=X20+X80-X50
              X3A=X30+X90-X60
              DATA(NDATA,14)=X1A
              DATA(NDATA,15)=X2A
              DATA(NDATA,16)=X3A
              PLA1=ATRIG(X10,X20,X30,X40,X50,X60,X70,X80,X90)
              PLA2=ATRIG(X10,X20,X30,X70,X80,X90,X1A,X2A,X3A)
              PLAREA(NDATA)=PLA1+PLA2
              WRITE(6,612) C1,C2,C3,CONST,PLAREA(NDATA)
```

```
      WRITE(6,333)
  112 IF(N1-1.0)113,113,100
  113 IF(N2-1.0)114,114,100
  114 IF(IPTION-5) 115,152,100
  115 IF(IPTION-4) 116,139,152
  116 NUMBER=1.0
      EXB1=B3*E2-B2*E3
      EXB2=B1*E3-B3*E1
      EXB3=B2*E1-B1*E2
      EXBMAG=(EXB1**2+EXB2**2+EXB3**2)**.5
      BXEXB1=B3*(B3*E1-B1*E3)-B2*(B1*E2-B2*E1)
      BXEXB2=B1*(B1*E2-B2*E1)-B3*(B2*E3-B3*E2)
      BXEXB3=B2*(B2*E3-B3*E2)-B1*(B3*E1-B1*E3)
      EXB4=EXB1/EXBMAG
      EXB5=EXB2/EXBMAG
      EXB6=EXB3/EXBMAG
      FPHI=PI*N1
      LOWLIM=0.0
      R=1.0
  117 GMAG=N2*L*(R-.5)
      UPLIM=R*N2*L
      IF(GMAG-L)118,100,100
  118 G1=GMAG*B1/BMAG
      G2=GMAG*B2/BMAG
      G3=GMAG*B3/BMAG
      IF(IPTION-1) 120,119,120
  119 RADIUS=RAD
      FGMAG=.5*N2*L
      AREA=4.0*FGMAG*RADIUS*FPHI
      GO TO 123
  120 BMGMAG=((B1-G1)**2+(B2-G2)**2+(B3-G3)**2)**.5
      IF ( IPTION -2)    122,121,122
  121 RADIUS=RAD*BMGMAG
      FPRR=FPHI*RAD*(1.0+RAD**2)**.5
      AREA=FPRR*(UPLIM-LOWLIM)*(2.0*BMAG-UPLIM-LOWLIM)
      LOWLIM=UPLIM
      GO TO 123
  122 RADIUS=SQRT(RAD*BMGMAG)
      RAG=RAD**2-4.0*ABS(RAD)*(LOWLIM-BMAG)
      RUG = RAD**2 - 4.0*ABS(RAD)*(UPLIM - BMAG)
      FPRAD = FPHI/(6.0*RAD)
      AREA =FPRAD*(RAG**1.5 - RUG**1.5)
      LOWLIM = UPLIM
  123 M= 1.0
  124 PHI = (2.0*M - 1.0)*N1*PI
      IF(PHI - TWOPI ) 125,138,138
  125 N = ABS( 1./TAN(PHI) )
      BXEXBM = ( (BXEXB1**2 + BXEXB2**2 + BXEXB3**2)**.5 )/N
      BXEXB4 = BXEXB1 / BXEXBM
      BXEXB5 = BXEXB2 / BXEXBM
      BXEXB6 = BXEXB3 / BXEXBM
      RHO = RADIUS  /   (N**2 + 1.0 )**.5
      RVXP = RHO * (EXB4 + BXEXB4 )
      RVYP=RHO* (EXB5+BXEXB5)
      RVZP=RHO* (EXB6+BXEXB6)
      RVXM=RHO* (EXB4-BXEXB4)
```

```
      RVYM=RHO* (EXB5-BXEXB5)
      RVZM=RHO* (EXB6-BXEXB6)
      RVXBEB=RADIUS* BXEXB4/N
      RVYBEB=RADIUS* BXEXB5/N
      RVZBEB=RADIUS* BXEXB6/N
      RVXEB=RADIUS*EXB4
      RVYEB=RADIUS*EXB5
      RVZEB=RADIUS*EXB6
      IF(PHI)126,126,127
  126 WRITE(6,215) PHI
      GO TO 100
  127 IF (PHI - HALFPI) 128,129,130
  128 RVX = RVXP
      RVY = RVYP
      RVZ = RVZP
      GO TO 137
  129 RVX = RVXEB
      RVY = RVYEB
      RVZ = RVZEB
      GO TO 137
  130 IF (PHI-PI      ) 131,132,133
  131 RVX = RVXM
      RVY = RVYM
      RVZ = RVZM
      GO TO 137
  132 RVX = -RVXBEB
      RVY = -RVYBEB
      RVZ = -RVZBEB
      GO TO 137
  133 IF (PHI -PI*1.5)  134,135,136
  134 RVX = -RVXP
      RVY = -RVYP
      RVZ = -RVZP
      GO TO 137
  135 RVX = -RVXEB
      RVY = -RVYEB
      RVZ = -RVZEB
      GO TO 137
  136 RVX = -RVXM
      RVY = -RVYM
      RVZ = -RVZM
  137 VP1 = X40 - G1 + RVX
      VP2 = X50 - G2 + RVY
      VP3 = X60 - G3 + RVZ
      WRITE(6,214)NUMBER, VP1,VP2,VP3,AREA
      NFIX=NUMBER
      VECTOR(NDATA,NFIX,1)=VP1
      VECTOR(NDATA,NFIX,2)=VP2
      VECTOR(NDATA,NFIX,3)=VP3
      VAREA(NDATA,NFIX)=AREA
      NUMBER=NUMBER+1.0
      M=M+1.0
      GO TO 124
  138 R = R + 1.0
      IF(GMAG - L)117,100,100
  139 NUMBER =1.0
```

```
      FTHETA = TWOPI *N2
      PHIO = 0.0
      F = 1.0
  140 Z = 1.0
      PHI1 = (2.0*F - 1.0) *PI*N1
      IF (PHI1 - TWOPI ) 141,100,100
  141 THETA1=TWOPI *(2.0*Z-1.0)*N2
      PHI2=  PI  *F* N1
      DARI=RMAG**2*(COS(PHIO))**2+BMAG**2*(SIN(PHIO))**2
      DARF=RMAG**2* (COS(PHI2))**2+BMAG**2* (SIN(PHI2))**2
      LOWLIM=BMAG-RMAG*BMAG*COS(PHIO)/SQRT(DARI)
      UPLIM=BMAG-RMAG*BMAG*COS(PHI2)/SQRT(DARF)
      A=(BMAG*RMAG)**2
      B=2.0*BMAG*(BMAG**2-RMAG**2)
      C=RMAG**2-BMAG**2
  161 IF (ABS(BMAG-RMAG).LT.1.0E-10) GO TO 143
      RFTBSC=RMAG*FTHETA/(4.0*BMAG**2*C)
      RFTABC=RMAG*FTHETA*(4.0*A*C-B**2)/(8.0*BMAG**2*C)
      CXFB=2.0*C* UPLIM+B
      CXIB=2.0*C* LOWLIM+B
      RTFL=SQRT(A+B*UPLIM+C*UPLIM**2)
      RTIL=SQRT(A+B*LOWLIM+C*LOWLIM**2)
      FAREA=RFTBSC*(CXFB*RTFL-CXIB*RTIL)
      IF (BMAG-RMAG) 142,143,144
  142 CC=C
      NUMARG=RTFL+UPLIM*SQRT(CC)+B/(2.0*SQRT(CC))
      DENARG=RTIL+LOWLIM*SQRT(CC)+B/(2.0*SQRT(CC))
      SAREA=(1.0/SQRT(CC))*ALOG(NUMARG/DENARG)
      GO TO 145
  143 AREA = RMAG * FTHETA * (UPLIM - LOWLIM)
      GO TO 146
  144 ASFARG=-(2.0*C*UPLIM+B)/SQRT( B**2 - 4.*A*C )
      ASIARG=-(2.0*C*LOWLIM+B)/ SQRT( B**2 - 4.*A*C )
      CC=-C
      FAREA=(1.0/SQRT(CC))*(ASIN(ASFARG)-ASIN(ASIARG))
  145 AREA = FAREA + RFTABC*SAREA
  146 DARO = RMAG**2 * COS(PHI1/2.)**2 + BMAG**2 * SIN(PHI1/2.)**2
      D=BMAG*RMAG/SQRT(DARO)
      A1=D*SIN(PHI1/2.0)*R1/RMAG
      A2=D*SIN(PHI1/2.0)*R2/RMAG
      A3=D*SIN(PHI1/2.0)*R3/RMAG
      DANG=D**2*(SIN(PHI1/2.0))**2*COS(THETA1/2.0)
      EPS=1.0E-10
      IF (ABS(A1).LT.EPS.AND.ABS(A2).LT.EPS) GO TO 301
      IF (ABS(B1).LT.EPS.AND.ABS(B2).LT.EPS) GO TO 302
      DET=A1*B2-A2*B1
      S=B2*DANG/DET
      T=(A2*B3-A3*B2)/DET
      U=-B1*DANG/DET
      V=(A3*B1-A1*B3)/DET
      P=T**2+V**2+1.0
      Q=2.0*(S*T+U*V)
      R=S**2+U**2-(D*SIN(PHI1/2.0))**2
      IF(THETA1 - TWOPI )147,147,148
  147 QD = 1.0
      GO TO 149
```

```
  148 QD =-1.0
  149 G3 =(-Q + QD * (Q**2 - 4.0*P*R )**.5 )/(2.0 * P)
      G2=U+V*G3
      G1=S+T*G3
      GO TO 320
  301 G3=DANG/A3
      FF1=B1
      FF2=B2
      HH1=-B3*G3
      GO TO 305
  302  G3=0.0
      FF1=A1
      FF2=A2
      HH1=DANG
  305 HH2=(D*SIN(PHI1/2.0))**2-G3**2
      IF (THETA1-TWOPI ) 306,306,307
  306 QD=1.0
      GO TO 309
  307 QD=-1.0
  309 IF (ABS(FF1).LT.EPS) GO TO 311
      COEF0=FF1**2+FF2**2
      COEF1=FF2*HH1/COEF0
      COEF2=(HH1**2-HH2*FF1**2)/COEF0
      G2=COEF1+QD*SQRT(ABS(COEF1**2-COEF2))
      G1=(HH1-FF2*G2)/FF1
      GO TO 320
  311 G2=HH1/FF2
      G1=QD*SQRT(ABS(HH2-G2**2))
  320 CONTINUE
      X41=2.0*X10-X40
      X51=2.0*X20-X50
      X61=2.0*X30-X60
      AXIMAG=1.0-D*COS(PHI1/2.0)/BMAG
      AXI1=AXIMAG*B1
      AXI2=AXIMAG*B2
      AXI3=AXIMAG*B3
      VP1=X41+AXI1+G1
      VP2=X51+AXI2+G2
      VP3=X61+AXI3+G3
      Z=Z+1.0
      IF(THETA1-2.0*TWOPI)150,151,151
  150 WRITE(6,214)NUMBER, VP1,VP2,VP3,AREA
      NFIX=NUMBER
      VECTOR(NDATA,NFIX,1)=VP1
      VECTOR(NDATA,NFIX,2)=VP2
      VECTOR(NDATA,NFIX,3)=VP3
      VAREA(NDATA,NFIX)=AREA
      NUMBER = NUMBER + 1.0
      GO TO 141
  151 PHI0 = PHI2
      F = F +1.0
      GO TO 140
  152 NUMBER = 1.0
      BDOTE = B1*E1+B2*E2+B3*E3
      AREA=N1*N2*SQRT((BMAG*EMAG)**2-BDOTE**2)
      HOR=1.0/N1
```

```
      VERT=1.0/N2
      V=1.0
  153 H=1.0
  154 Q1=N1*(H-.5)*B1
      Q2=N1*(H-.5)*B2
      Q3=N1*(H-.5)*B3
      W1=N2*(V-.5)*E1
      W2=N2*(V-.5)*E2
      W3=N2*(V-.5)*E3
      VP1=X40+W1-Q1
      VP2=X50+W2-Q2
      VP3=X60+W3-Q3
      WRITE(6,214)NUMBER, VP1,VP2,VP3,AREA
      NFIX=NUMBER
      VECTOR(NDATA,NFIX,1)=VP1
      VECTOR(NDATA,NFIX,2)=VP2
      VECTOR(NDATA,NFIX,3)=VP3
      VAREA(NDATA,NFIX)=AREA
      NUMBER=NUMBER+1.0
      H=H+1.0
      IF(H-HOR)154,154,155
  155 V = V+1.0
      IF(V-VERT)153,153,100
  100 CONTINUE
  601 IF (IPTION.LE.5)                                        GO TO 699
      CALL DISK (0,NDATA,PDUM,XDUM,XTDUM,IHIT)
  690 GO TO 500
  699 CONTINUE
      IF (IPTION.EQ.5) GO TO 501
      COEF(NDATA,1)=C1
      COEF(NDATA,2)=C2
      COEF(NDATA,3)=C3
      COEF(NDATA,4)=C4
      COEF(NDATA,5)=C5
      COEF(NDATA,6)=C6
      COEF(NDATA,7)=C7
      COEF(NDATA,8)=C8
      COEF(NDATA,9)=C9
      COEF(NDATA,10)=CONST
      GO TO 503
  501 DO 502 IZERO=1,6
  502 COEF(NDATA,IZERO)=0.0
      COEF(NDATA,7)=C1
      COEF(NDATA,8)=C2
      COEF(NDATA,9)=C3
      COEF(NDATA,10)=-CONST
  503 CONTINUE
      IF (IPTION.GT.3) GO TO 500
      DO 360 KK=1,2
      IF (KK.EQ.1 .AND. IPTION.GT.2)                          GO TO 360
      WRITE(6,301) KK
      IF (KK.EQ.2) GO TO 351
      NRG=NRING1
      GO TO 352
  351 NRG=NRING2
  352 DO 370 JJ=1,NRG
```

```
      WRITE(6,362) JJ,RMEANC(NDATA,JJ,KK),CRING(NDATA,JJ,KK)
  370 CONTINUE
  360 CONTINUE
  500 CONTINUE
      GO TO 10
  201 FORMAT (1H1,1X,25HINPUT DATA FOR TARGET NO. ,I3 /)
  202 FORMAT (/25H TRANSFORMATION MATRIX        )
  203 FORMAT ( 5E16.6 )
  204 FORMAT(/30H COEFFICIENTS FOR A CYLINDER       )
  205 FORMAT(/30H COEFFICIENTS FOR A CONE           )
  206 FORMAT(/35H COEFFICIENTS FOR A PARABOLOID          )
  207 FORMAT(   6H C1 = , E12.6 / 6H C2 =  , E12.6 / 6H C3 =  , E12.6 /
     1 6H C4 =  , E12.6 / 6H C5 =  , E12.6 / 6H C6 =  , E12.6 / 6H C7 =
     2  , E12.6 / 6H C8 =  , E12.6 / 6H C9 =  , E12.6 /  9H CONST =  ,
     3 E12.6,/)
  607 FORMAT(   6H C1 = , E12.6 / 6H C2 =  , E12.6 / 6H C3 =  , E12.6 /
     1 6H C4 =  , E12.6 / 6H C5 =  , E12.6 / 6H C6 =  , E12.6 / 6H C7 =
     2  , E12.6 / 6H C8 =  , E12.6 / 6H C9 =  , E12.6 /  9H CONST =  ,
     3  E12.6 / 30H RBOND(NDATA), RBONDU(NDATA)   , 2E12.6 / )
  208 FORMAT(/40H COEFFICIENTS FOR CONSTRAINT PLANES          )
  209 FORMAT(   7H C11 =  , E12.6 / 7H C12 =  , E12.6 / 7H C13 =  ,
     1 E12.6 / 9H CEND  =  ,   E12.6 /  8H CBASE=   , E12.6   )
  210 FORMAT(/35H COEFFICIENTS FOR AN ELLIPSOID          )
  211 FORMAT(/30H COEFFICIENTS FOR A PLANE               )
  212 FORMAT(   6H C1 =  , E12.6 /  6H C2 =  , E12.6 / 6H C3 =  , E12.6
     1  /  9H CONST =  , E12.6)
  612 FORMAT(   6H C1 =  , E12.6 /  6H C2 =  , E12.6 / 6H C3 =  , E12.6
     1  /  9H CONST =  , E12.6,/,24H AREA OF PARALLELOGRAM =,E12.6)
  214 FORMAT(3X,F5.0,5X,4HVP1=,E12.6,5X,4HVP2=,E12.6,5X,4HVP3=,E12.6,
     * 8X,5HAREA=,E12.6)
  215 FORMAT( / 7H PHI =   , E15.5 )
  251 FORMAT (5I4,4X,7F8.4)
  252 FORMAT (10F8.4)
  333 FORMAT  (/,1X,35HCOMPONENTS FOR VIEW POINT VECTORS  )
  941 FORMAT(1X,36HREFERENCE POINTS ON CONSTRAINT DISK ,3X,
     * 2HT(,3F10.5,2H ),5X,2HS(,3F10.5,2H ),/)
  361 FORMAT(/,23H ON CONSTRAINT DISK NO.,I1,/,12X,7H RMEAN ,15X,4HAREA)
  362 FORMAT(I5,3X,E12.6,8X,E12.6)
      END
```

@PRT,S TRANSF

```
LLL-ALBIN202*TPFS.TRANSF
          SUBROUTINE TRANSF (X,XP,V,VP,IN,IOUT)
      C
      C --- THIS PROGRAM TRANSFORMS INPUT IN X-COORD SYSTEM TO X-PRIME SYSTEM.
      C --- IN AND IOUT REFER TO COORD SYSTEM. 0, CENTRAR SYSTEM.
      C --- 1, LEFT PLUME COORDINATES,  2, RIGHT PLUME COORDINATES.
      C --- SIGMA MEASURES BETWEEN X1-AXIS AND X1P-AXIS.
      C --- PSI MEASURES FROM X2-AXIS, ROTATES ABOUT X1-AXIS
      C --- X AND XP ARE POSITION VECTORS, V AND VP ARE VELOCITY VECTORS.
      C
          COMMON /TWOPLM/PP(2,3),SIG(2),PSI(2)
          DIMENSION X(3),XP(3),V(3),VP(3),XS(3),VS(3),PPP(3)
          IWRITE=0
          IPASS=0
          DO 50 I=1,3
          XP(I)=X(I)
          VP(I)=V(I)
          XS(I)=X(I)
       50 VS(I)=V(I)
          IF (IN.EQ.IOUT)                               GO TO 500
          IF (IN.EQ.0)                                  GO TO 100
          IF (IOUT.EQ.0)                                GO TO 200
          GO TO 300
      C --- TRASFORM FROM 0 TO EITHER 1 OR 2
      100 SIGMA=SIG(IOUT)
          PSIH=PSI(IOUT)
          DO 110 I=1,3
      110 PPP(I)=PP(IOUT,I)
          SINS=SIN(SIGMA)
          COSS=COS(SIGMA)
          SINP=SIN(PSIH)
          COSP=COS(PSIH)
      120 XP(1)=(X(1)-PPP(1))*COSS+(X(2)-PPP(2))*SINS*SINP
         1      -(X(3)-PPP(3))*SINS*COSP
          XP(2)=(X(2)-PPP(2))*COSP+(X(3)-PPP(3))*SINP
          XP(3)=(X(1)-PPP(1))*SINS-(X(2)-PPP(2))*COSS*SINP
         1      +(X(3)-PPP(3))*COSS*COSP
          VP(1)=V(1)*COSS+V(2)*SINS*SINP-V(3)*SINS*COSP
          VP(2)=V(2)*COSP+V(3)*SINP
          VP(3)=V(1)*SINS-V(2)*COSS*SINP+V(3)*COSS*COSP
          IPASS=IPASS+1
          GO TO 300
      C --- TRANSFORM FROM 1 OR 2 TO 0
      200 SIGMA=SIG(IN)
          PSIH=PSI(IN)
          DO 210 I=1,3
      210 PPP(I)=PP(IN,I)
          SINS=SIN(SIGMA)
          COSS=COS(SIGMA)
          SINP=SIN(PSIH)
          COSP=COS(PSIH)
      220 XP(1)= X(1)*COSS                   +X(3)*SINS      +PPP(1)
          XP(2)= X(1)*SINS*SINP+X(2)*COSP-X(3)*COSS*SINP+PPP(2)
          XP(3)=-X(1)*SINS*COSP+X(2)*SINP+X(3)*COSS*COSP+PPP(3)
          VP(1)= V(1)*COSS                   +V(3)*SINS
          VP(2)= V(1)*SINS*SINP+V(2)*COSP-V(3)*COSS*SINP
```

```
      VP(3)=-V(1)*SINS*COSP+V(2)*SINP+V(3)*COSS*COSP
      IPASS=IPASS+1
      GO TO 300
C --- TRANSFORM FROM 1 TO 2 AND VICE VERSA
  300 IF (IN.EQ.0 .OR. IOUT.EQ.0)                      GO TO 500
      IF (IPASS.EQ.0)                                  GO TO 200
      IF (IPASS.EQ.2)                                  GO TO 500
      DO 320 I=1,3
      X(I)=XP(I)
  320 V(I)=VP(I)
      GO TO 100
  500 CONTINUE
      DO 520 I=1,3
      X(I)=XS(I)
  520 V(I)=VS(I)
      IF (IWRITE.EQ.1)                                 GO TO 600
      RETURN
  600 WRITE (6,601) (X(I),I=1,3),(XP(I),I=1,3),(V(I),I=1,3),
     *                (VP(I),I=1,3),IN,IOUT
  601 FORMAT (7H0X XP =,2(3F8.4,2X),7H V VP =,2(3F8.4,2X),7HIN-OUT=,2I2)
      RETURN
      END
```

?PRT,S VFLHIT

```
LEE-ALBIN202*TPFS.VFEMIT
      SUBROUTINE VFEMIT (IO,X,A,EAREA)
C     IO=0, TO READ INPUT DATA OF THE EMITTING SURFACE
C     IO=1, TO GIVE LOCATION X AND DIRECTION A OF THE EMITTING RAY
      DIMENSION X(3),A(3),P1(3),P2(3),P3(3),DC(3),S(3)
      DATA PI,TWOPI,HALFPI/3.14159, 6.28318, 1.57079/
      RANGE(X1,X2,X3,Y1,Y2,Y3)=SQRT((X1-Y1)**2+(X2-Y2)**2+(X3-Y3)**2)
      MTX(X11,X12,X13,X21,X22,X23,X31,X32,X33) = X11*X22*X33+X21*X13*X32
     *                        +X12*X23*X31-X13*X22*X31-X11*X23*X32-X33*X12*X21
      IF (IO.NE.0)                                       GO TO 200
      NEMIT=3
      IWRITE=0
C     IO=0, INPUT PART
C     IEMIT IDENTIFIES THE EMITTING SURFACE, SAME AS IPTION IN TARGET
C *** IEMIT=1, HALF CYLINDER, 2, HALF FRUSTUM, 3,4, HEMISPHERE,
C ***          5, PARALLELOGRAM, 6, ANNULAR DISK,
C ***          7, FULL CYLINDER, 8, FULL FRUSTUM, 9, SPHERE.
  100 READ (5,191) IEMIT
      GO TO (110,120,130,140,150,160,110,120,130),IEMIT
C     P1 ON TOP, P2 ON BOTTOM, C IS FROM P2 TO P1
  110 READ (5,192) R1,(P1(I),I=1,3),(P2(I),I=1,3),(DC(I),I=1,3)
      H=RANGE(P1(1),P1(2),P1(3),P2(1),P2(2),P2(3))
      R2=R1
      RMEAN=R1
      RL=R2
  115 C1=(P1(1)-P2(1))/H
      C2=(P1(2)-P2(2))/H
      C3=(P1(3)-P2(3))/H
      EAREA=RMEAN*PI*H
      IF (IEMIT.GE.7)                                    EAREA=2.*EAREA
      GO TO 181
C     DC IS PERPENDICULAR TO P1-P2 AXIS, EXTENDING FROM P2 TO P3, R2=P3-P2.
  120 READ (5,192) R1,(P1(I),I=1,3),(P2(I),I=1,3),(P3(I),I=1,3)
      H=RANGE(P1(1),P1(2),P1(3),P2(1),P2(2),P2(3))
      R2=RANGE(P3(1),P3(2),P3(3),P2(1),P2(2),P2(3))
      DC(1)=(P3(1)-P2(1))/R2
      DC(2)=(P3(2)-P2(2))/R2
      DC(3)=(P3(3)-P2(3))/R2
      RMEAN=(R1+R2)/2.
      TVTX=(R2-R1)/H
      VTX=ATAN(TVTX)
      CVTX=COS(VTX)
      SVTX=SIN(VTX)
      GO TO 115
  130 CONTINUE
  140 READ (5,192) R1,(P1(I),I=1,3),(DC(I),I=1,3)
      EAREA=TWOPI*R1*R1
      IF (IEMIT.EQ.9)                                    EAREA=2.*EAREA
      GO TO 181
C     NORMAL IN RHR SENSE, P1-P2-P3.
  150 READ (5,192) (P1(I),I=1,3),(P2(I),I=1,3),(P3(I),I=1,3)
      H12=RANGE(P1(1),P1(2),P1(3),P2(1),P2(2),P2(3))
      H23=RANGE(P2(1),P2(2),P2(3),P3(1),P3(2),P3(3))
      EAREA=H12*H23
      C121=(P2(1)-P1(1))/H12
      C122=(P2(2)-P1(2))/H12
```

```
      C123=(P2(3)-P1(3))/H12
      C231=(P3(1)-P2(1))/H23
      C232=(P3(2)-P2(2))/H23
      C233=(P3(3)-P2(3))/H23
      VMAG2=(C122*C233-C123*C232)**2+(C231*C123-C233*C121)**2+
     1      (C121*C232-C122*C231)**2
      VMAG=SQRT(VMAG2)
      DC(1)=(C122*C233-C123*C232)/VMAG
      DC(2)=(C231*C123-C233*C121)/VMAG
      DC(3)=(C121*C232-C122*C231)/VMAG
      GO TO 181
C     P2 IS ON THE OUTER RIM, THEREFORE  R2=ABS(P2-P1). R1=INNER RADIUS.
  160 READ (5,192) R1,(P1(I),I=1,3),(P2(I),I=1,3),(DC(I),I=1,3)
      R2=RANGE(P2(1),P2(2),P2(3),P1(1),P1(2),P1(3))
      EAREA=(R2*R2-R1*R1)*PI
      GO TO 181
  181 WRITE (6,195) IEMIT, R1,R2,(DC(I),I=1,3),(P1(I),I=1,3),
     *              (P2(I),I=1,3),(P3(I),I=1,3)
      GO TO 500
  191 FORMAT (8I8)
  192 FORMAT (10F8.0)
  195 FORMAT (////31H EMITTING SURFACE           IEMIT =,I3,10X, 7HR1,R2 =,
     1          2F8.3 ,10X,14HNORMAL D.C. = ,3F8.4 // 10X, 5HP1 = ,3F8.4,
     2          5X,5HP2 = ,3F8.4,5X,5HP3 = , 3F8.4  ////)
C
C *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** ***
C     IO=1. OUTPUT PART
C
  200 CONTINUE
      CALL RANDOM (RN1)
      CALL RANDOM (RN2)
      GO TO (210,220,230,240,250,260,210,220,230),IEMIT
  210 CONTINUE
      HELV=RN1*H
      GO TO 221
  219 CALL RANDOM (RN1)
  220 CONTINUE
      RL=R2*SQRT(RN1)
      IF (RL.LT.R1)                                      GO TO 219
      HELV=H*(R2-RL)/(R2-R1)
  221 IF (IEMIT.EQ.1 .OR. IEMIT.EQ.2)                    TH=(0.5-RN2)*HALFPI
      IF (IEMIT.EQ.7 .OR. IEMIT.EQ.8)                    TH=TWOPI*RN2
      COSTH=COS(TH)
      SINTH=SIN(TH)
      XM1=C3*DC(2)-C2*DC(3)
      XM2=C1*DC(3)-C3*DC(1)
      XM3=C2*DC(1)-C1*DC(2)
      DSCM=MTX(C1,C2,C3,DC(1),DC(2),DC(3),XM1,XM2,XM3)
      IF (ABS(DSCM).LT.1.E-6)
     *WRITE (6,302) IEMIT,DSCM,C1,C2,C3,DC(1),DC(2),DC(3),XM1,XM2,XM3
      SP1=MTX(0.,C2,C3,COSTH,DC(2),DC(3),SINTH,XM2,XM3)/DSCM
      SP2=MTX(C1,0.,C3,DC(1),COSTH,DC(3),XM1,SINTH,XM3)/DSCM
      SP3=MTX(C1,C2,0.,DC(1),DC(2),COSTH,XM1,XM2,SINTH)/DSCM
      X(1)=P2(1)+HELV*C1+RL*SP1
      X(2)=P2(2)+HELV*C2+RL*SP2
      X(3)=P2(3)+HELV*C3+RL*SP3
```

```
      IF (IEMIT.EQ.2 .OR. IEMIT.EQ.8)                      GO TO 225
      S(1)=SP1
      S(2)=SP2
      S(3)=SP3
      GO TO 290
  225 XM1=C2*SP3-C3*SP2
      XM2=C3*SP1-C1*SP3
      XM3=C1*SP2-C2*SP1
      DSCM=MTA(SP1,SP2,SP3,C1,C2,C3,XM1,XM2,XM3)
      IF (ABS(DSCM).LT.1.E-6)
     *WRITE (6,302) IEMIT,DSCM,C1,C2,C3,DC(1),DC(2),DC(3),XM1,XM2,XM3
      S(1)=MTA(CVTX,SP2,SP3,SVTX,C2,C3,0.,XM2,XM3)/DSCM
      S(2)=MTA(SP1,CVTX,SP3,C1,SVTX,C3,XM1,0.,XM3)/DSCM
      S(3)=MTA(SP1,SP2,CVTX,C1,C2,SVTX,XM1,XM2,0.)/DSCM
      GO TO 290
  230 CONTINUE
  240 CONTINUE
  241 COSANG=0.
      CALL DIFVDC (DC,TWOPI,A)
      DO 242 I=1,3
      X(I)=P1(I)+R1*A(I)
      S(I)=A(I)
  242 COSANG=COSANG+DC(I)*S(I)
      IF (IEMIT.EQ.9)                                      GO TO 290
      IF (COSANG.LT.0.)                                    GO TO 241
      GO TO 290
  250 CONTINUE
      D1=RN1*H12
      D2=RN2*H23
      X(1)=P1(1)+C121*D1+C231*D2
      X(2)=P1(2)+C122*D1+C232*D2
      X(3)=P1(3)+C123*D1+C233*D2
      GO TO 264
  260 CONTINUE
      DC1=(P2(1)-P1(1))/R2
      DC2=(P2(2)-P1(2))/R2
      DC3=(P2(3)-P1(3))/R2
  261 RR=R2*SQRT(RN1)
      IF (RR.GE.R1)                                        GO TO 262
      CALL RANDOM (RN1)
      GO TO 261
  262 TH=TWOPI*RN2
      COSTH=COS(TH)
      SINTH=SIN(TH)
      XM1=DC(2)*DC3-DC(3)*DC2
      XM2=DC(3)*DC1-DC(1)*DC3
      XM3=DC(1)*DC2-DC(2)*DC1
      DSCM=MTX(DC(1),DC(2),DC(3),DC1,DC2,DC3,XM1,XM2,XM3)
      IF (ABS(DSCM).LT.1.E-6)
     *WRITE (6,302) IEMIT,DSCM,DC(1),DC(2),DC(3),DC1,DC2,DC3,XM1,XM2,XM3
      B1=MTX(0.,DC(2),DC(3),COSTH,DC2,DC3,SINTH,XM2,XM3) / DSCM
      B2=MTX(DC(1),0.,DC(3),DC1,COSTH,DC3,XM1,SINTH,XM3) / DSCM
      B3=MTX(DC(1),DC(2),0.,DC1,DC2,COSTH,XM1,XM2,SINTH) / DSCM
      X(1)=P1(1)+B1*RR
      X(2)=P1(2)+B2*RR
      X(3)=P1(3)+B3*RR
```

```
      264 DO 265 I=1,3
      265 S(I)=DC(I)
      290 CALL DIFVDC (S,HALFPI,A)
          IF (IWRITE.EQ.0)                                    GO TO 300
          NEMIT=NEMIT+1
          IF (NEMIT.GT.100)                                   GO TO 300
          WRITE (6,301) NEMIT,(X(I),I=1,3),(A(I),I=1,3)
      300 CONTINUE
      301 FORMAT (13H NEMIT X A ,I12,2(4X,3F8.4)  )
      302 FORMAT (/28H IEMIT, DSCM, C11 C12 SO ON  ,I5,E14.5//10X,9E12.5 /)
      500 RETURN
          END

@PRT,S VFOUTP
```

```
LEE-ALBIN202*TPFS.VFOUTP
      SUBROUTINE VFOUTP (NRAY,NSTART,MHITP,MMISS,MHITG,EAREA)
C     THIS PROGRAM OUTPUT THE VIEW FACTORS AND ITS RECIPROCITY
      INCLUDE GEOM,LIST
   10 FORMAT (1H1,50X,18HVIEW FACTOR OUTPUT  )
   11 FORMAT (51X,18H------------------  //  )
   12 FORMAT (/9X,14HTOTAL SAMPLE =,I8, 8X,8HNSTART =,I7,10X, 7HMHITP =,
     1        I7, 5X,7HMMISS =,I7, 5X, 7HMHITG =,I7  //
     2        9X,7HEAREA =,1PE12.6  // )
   13 FORMAT (1H1)
   20 FORMAT (// 11H TARGET NO. , I3  /)
   21 FORMAT (30H TARGET MAIN SURFACE,       L = , I3   /)
   22 FORMAT (43H ON CONSTRAINT DISK PASSING THRU P1,    L =,I3 /)
   23 FORMAT (43H ON CONSTRAINT DISK PASSING THRU P2,    L =,I3 /)
   30 FORMAT (/4X,4HAREA,5X,6HNO. OF,5X,4HAREA,8X,4HVIEW,7X,7HHITAREA,
     1        6X,4HAREA,6X,14HVIEW FACTOR BY / 3X,6HNUMBER,5X,4HHITS,5X,
     2        6HNUMBER,6X,6HFACTOR,18X,6HNUMBER,7X,11HRECIPROCITY  /)
   31 FORMAT (4X,I3,5X,F6.0,6X,I3,7X,1PE9.4,3X,E9.4,5X,I3,9X,E10.4)
   32 FORMAT (/5X,12HTOTAL HIT = ,F8.0,8X,13HVIEW FACTOR =,1PE11.4,8X,
     *        11HRECIPR VF = ,E11.4 //)
   33 FORMAT ( / )
      WRITE (6,10)
      WRITE (6,11)
      WRITE (6,12) NRAY,NSTART,MHITP,MMISS,MHITG,EAREA
      DO 190 K=1,NTARGT
      NCLOCK=DATA(K,1)
      DO 180 L=1,3
      IF (IBODY(K).GT.3.AND.L.NE.1)                   GO TO 180
      IF (IBODY(K).GT.2.AND.L.EQ.2)                   GO TO 180
      WRITE (6,20) K
      GO TO (111,112,113),                            L
  111 WRITE (6,21) L
      GO TO 120
  112 WRITE (6,22) L
      GO TO 120
  113 WRITE (6,23) L
      GO TO 120
  120 CONTINUE
      WRITE (6,30)
      NA=NAREA(K,L)
      TOTA=0.
      CLINE=0.5
      DO 170 J=1,NA
      GO TO (131,132,132),                            L
  131 HTAREA=VAREA(K,J)
      GO TO 135
  132 NCHECK=(J-1)/NCLOCK+1
      HTAREA=CRING(K,NCHECK,L-1)
  135 CONTINUE
      TOTA=TOTA+HTAREA
      IF (HTAREA.LT.1.E-8)                            GO TO 170
      VF=HIT(J,K,L)/FLOAT(NRAY)
      RVF=VF*EAREA/HTAREA
      WRITE (6,31) J,HIT(J,K,L),J,VF,HTAREA,J,RVF
      CLINE=CLINE+1.0
      IF (CLINE.LT.DATA(K,1))                         GO TO 170
```

```
      WRITE (6,33)
      CLINE=0.5
  170 CONTINUE
      TOTHIT=0.
      DO 175 J=1,NA
  175 TOTHIT=TOTHIT+HIT(J,K,L)
      TVF=TOTHIT/FLOAT(NRAY)
      TRVF=TVF*EAREA/TOTA
      WRITE (6,32) TOTHIT,TVF,TRVF
  180 CONTINUE
      WRITE (6,13)
  190 CONTINUE
  300 CONTINUE
      RETURN
      END
```

PRT,S ZCOORD

```
LEE-ALBIN202*TPFS.ZCOORD
              SUBROUTINE ZCOORD(RIS,RI,J,H,HI,JZ)
              INCLUDE DIM,LIST
C CALCULATE Z-COORDINATE OF INTERSECTION OF PHOTON PATH WITH CONE
              ALPHA=THETA-PHII
              IF (ALPHA.LT.0.)                                  ALPHA=ALPHA+TWOPI
              COST=COS(ALPHA)
            4 W = H * TANETA
              W2 = W**2
              RR = RIS + W2 - 2.*RI*W*COST
              R = SQRT(RR)
              TANBTP = (RR + W2 - RIS)/(2.*H       *R)
              TNETA2 = TANETA**2
              Q = R*TANBTP
              Q2 = Q**2
              IF(COST.LT.0.0) GO TO 120
C BUNDLE INTERSECTS ONLY ONE SIDE OF CONE ARRAY
              IF(COSETA.GE.CGAMMA(JRGN)) GO TO 150
          156 L = JNDEX
              J2 = 1
              J3 = JRGN - JNDEX + 1
              GO TO 151
          150 L = 1
              DO 152 JZ = 1,JRGN
          152 IF( COSETA . GT . CGAMMA(JZ) ) GO TO 153
          153 IF(JZ-JNDEX) 154,65,156
          154 J1 = JNDEX - JZ
              DO 155 J=1,J1
              Y=TNETA2-TANG2(JNDEX-J)
              Z1P = SQRT(ABS(Q2-RR*Y))
          155 Z(J) = (Q-Z1P)/Y
              Z(J1+1)= HZ(IRGN+1)+1.0
              GO TO 66
C CHECK FOR DOUBLE INTERSECTIONS WITH CONE ARRAY
          120 CA = RIS * ( 1. - COST ** 2 )
              IF ( COSETA . LT . 0.0 ) GO TO 141
              DO  122 JZ = 1,JRGN
              IF ( COSETA . GE . CGAMMA(JZ) ) GO TO 123
              CB = CA / ( TNETA2 - TANG2(JZ) )
              X = SQRT( CB * TANG2(JZ) )
              ZA = X * TANETA / TANG2(JZ)
              ZZ = ( X - RI * COST ) / TANETA +H
          122 IF ( ZZ . GT . ZA) GO TO 123
              GO TO 123
          141 DO 142 JZ = 1,JRGN
              IF ( JZ . EQ . JNDEX ) GO TO 123
              IF ( -COSETA . GE . CGAMMA(JZ) ) GO TO 142
              CB = CA / ( TNETA2 - TANG2(JZ) )
              X = -SQRT( CB * TANG2(JZ) )
              ZA = X * TANETA / TANG2(JZ)
              ZZ = ( X - RI * COST ) / TANETA + H
              IF ( ZZ . GT . ZA ) GO TO 123
          142 CONTINUE
          123 J1 = JNDEX - JZ
              IF (J1 .EQ. 0  ) GO TO 125
C BUNDLE PATH POSSIBLY TWO INTERSECTIONS WITH CONES
```

```
          DO 124 J = 1,J1
          L = JNDEX - J
          Y = TNETA2 - TANG2(L)
          ZIP = SQRT(ABS(Q2-RR*Y))
          IF(COSETA.LT.0.0) GO TO 131
          Z(J) = (Q-ZIP) / Y
          GO TO 124
      131 Z(J) = (Q+ZIP) / Y
      124 CONTINUE
C BUNDLE INTERSECTS ONLY ONE SIDE OF CONE ARRAY
      125 J2 = J1+1
          J3 = J2 + JRGN - J2
          L = J2
      151 DO 126 J=J2,J3
          IF ( CGAMMA(L) . LE . COSETA ) GO TO 47
          Y = TNETA2 - TANG2(L)
          ZIP = SQRT(ABS(Q2-RR*Y))
          IF(COSETA.LT.0.0) GO TO 127
          Z(J) = (Q + ZIP)/Y
          GO TO 126
      127 Z(J) = (Q - ZIP)/Y
      126 L = L+1
          GO TO 66
       65 Z(L) = HZ(IRGN+1) + 1.0
          GO TO 66
       47 Z(J) = HZ(IRGN+1) + 1.0
       66 J = 1
          RETURN
          END
```

```
@ELT,L FORCARDS
ELT007 RLIB70 03/05-11:43:08-(0,)
000001        000    @FOR,S ATTEN,ATTEN
000002        000    @FOR,S DISK,DISK
000003        000    @FOR,S EMITT,EMITT
000004        000    @FOR,S ESCAP,ESCAP
000005        000    @FOR,S ESCAPE,ESCAPE
000006        000    @FOR,S FRTAPE,FRTAPE
000007        000    @FOR,S INPUT,INPUT
000008        000    @FOR,S INTRCP,INTRCP
000009        000    @FOR,S MAIN,MAIN
000010        000    @FOR,S MAINS,MAINS
000011        000    @FOR,S OUTPUT,OUTPUT
000012        000    @FOR,S QUICKP,QUICKP
000013        000    @FOR,S SCATTR,SCATTR
000014        000    @FOR,S SORTNG,SORTNG
000015        000    @FOR,S TARGET,TARGET
000016        000    @FOR,S VFOUTP,VFOUTP
000017        000    @FOR,S ZCOORD,ZCOORD

END ELT.
```

```
@ELT,L PUNCARDS
ELT007 RLIB70 03/05-11:43:09-(0,)
000001      000   @PCH,S DIMENS
000002      000   @PCH,S ATTEN
000003      000   @PCH,S CHOSE
000004      000   @PCH,S DIFVDC
000005      000   @PCH,S DISK
000006      000   @PCH,S EMITT
000007      000   @PCH,S ESCAP
000008      000   @PCH,S ESCAPE
000009      000   @PCH,S FRTAPE
000010      000   @PCH,S INPUT
000011      000   @PCH,S INTRCP
000012      000   @PCH,S IOPKT
000013      000   @PCH,S MAIN
000014      000   @PCH,S MAINS
000015      000   @PCH,S OUTPUT
000016      000   @PCH,S PINGEA
000017      000   @PCH,S PINGEB
000018      000   @PCH,S QUADEQ
000019      000   @PCH,S QUICKP
000020      000   @PCH,S SCATTR
000021      000   @PCH,S SORTNG
000022      000   @PCH,S SPHERE
000023      000   @PCH,S TARGET
000024      000   @PCH,S TRANSF
000025      000   @PCH,S VFEMIT
000026      000   @PCH,S VFOUTP
000027      000   @PCH,S ZCOORD
END ELT.

@PRT,T
FURPUR 0026-03/05-11:43
```

LEE-ALBIN202*TPF$ ELEMENT TABLE

D NAME	VERSION	TYPE	DATE	TIME	SEQ #	SIZE-PRE	TEXT	(CYCLE	WORD)	PSRMODE	LOCATION
PINGEA		FOR SYMB	18 MAR 75	21:15:11	1		17	5	0	1	1792
CHOSE		FOR SYMB	18 MAR 75	21:15:14	2		9	5	0	1	1809
QUADEQ		FOR SYMB	18 MAR 75	21:15:16	3		9	5	0	1	1818
PINGEB		FOR SYMB	18 MAR 75	21:15:18	4		10	5	0	1	1827
EMITT		FOR SYMB	20 AUG 75	00:07:24	5		7	0	0	1	1837
RANDOM		FOR SYMB	15 OCT 75	14:36:01	6		3	5	0	1	1844
QUICKP		FOR SYMB	15 OCT 75	14:36:36	7		16	5	0	1	1847
DISK		FOR SYMB	31 OCT 75	11:09:31	8		23	5	0	1	1863
MAINS		FOR SYMB	21 NOV 75	02:20:28	9		26	0	0	1	1886
ATTEN		FOR SYMB	21 NOV 75	02:20:33	10		48	0	0	1	1912
ZCOORD		FOR SYMB	21 NOV 75	02:20:39	11		19	0	0	1	1960
CHOSE		RELOCATABLE	23 DEC 75	20:16:07	12	1	8				1979
PINGEA		RELOCATABLE	23 DEC 75	20:18:45	13	1	20				1988
PINGEB		RELOCATABLE	23 DEC 75	20:18:47	14	1	11				2009
QUADEQ		RELOCATABLE	23 DEC 75	20:18:49	15	1	12				2021
SPHERE		FOR SYMB	07 JAN 76	23:01:28	16		191	5	0	1	2034

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

SPHERE	RELOCATABLE	07 JAN 76	23:01:36	17	2	62				2225
IOPKT	FOR SYMB	16 JAN 76	00:13:59	18		8	5	0	1	2289
IOPKT	RELOCATABLE	16 JAN 76	00:14:00	19	2	9				2297
PUNCARDS	ELT SYMB	19 JAN 76	12:13:34	20		4	5	0	1	2308
ESCAP	FOR SYMB	19 JAN 76	12:13:37	21		8	5	0	1	2312
INTRCP	FOR SYMB	19 JAN 76	12:13:53	22		12	5	0	1	2320
PRTCARDS	ELT SYMB	26 JAN 76	10:50:40	23		4	5	0	1	2332
FORCARDS	ELT SYMB	26 JAN 76	10:50:40	24		3	5	0	1	2336
ESCAPE	FOR SYMB	26 JAN 76	10:50:45	25		16	5	0	1	2339
RANDOM	RELOCATABLE	26 JAN 76	10:53:38	26	1	4				2355
MAIN	FOR SYMB	27 JAN 76	15:30:01	27		28	5	0	1	2360
DIMENS	FOR PROC	29 JAN 76	00:34:52	28		11	1	0	1	2388
ATTEN	RELOCATABLE	29 JAN 76	00:34:57	29	4	43				2399
DISK	RELOCATABLE	29 JAN 76	00:35:01	30	2	27				2446
EMITT	RELOCATABLE	29 JAN 76	00:35:04	31	4	8				2475
ESCAP	RELOCATABLE	29 JAN 76	00:35:07	32	4	7				2487
ESCAPE	RELOCATABLE	29 JAN 76	00:35:10	33	5	20				2498
INTRCP	RELOCATABLE	29 JAN 76	00:35:26	34	4	12				2523
MAIN	RELOCATABLE	29 JAN 76	00:35:30	35	6	24				2539
MAINS	RELOCATABLE	29 JAN 76	00:35:33	36	5	32				2569
OUTCRP	RELOCATABLE	29 JAN 76	00:35:42	37	4	19				2606
ZCOORD	RELOCATABLE	29 JAN 76	00:36:09	38	4	19				2629
DIFVOC	FOR SYMB	03 FEB 76	20:39:45	39		8	5	0	1	2652
DIFVOC	RELOCATABLE	03 FEB 76	20:39:47	40	2	11				2660
OUTPUT	FOR SYMB	06 FEB 76	06:30:59	41		36	5	0	1	2673
OUTPUT	RELOCATABLE	06 FEB 76	06:31:02	42	4	45				2709
INPUT	FOR SYMB	12 FEB 76	01:58:22	43		81	5	0	1	2758
INPUT	RELOCATABLE	12 FEB 76	01:58:30	44	4	77				2839
SCATTR	FOR SYMB	12 FEB 76	01:58:35	45		13	5	0	1	2920
SCATTR	RELOCATABLE	12 FEB 76	01:58:39	46	4	15				2933
VFEMIT	FOR SYMB	12 FEB 76	01:58:40	47		47	5	0	1	2952
VFEMIT	RELOCATABLE	12 FEB 76	01:58:46	48	2	58				2999
VFOUTP	FOR SYMB	12 FEB 76	01:58:46	49		18	5	0	1	3059
VFOUTP	RELOCATABLE	12 FEB 76	01:58:50	50	2	21				3077
SORTNG	FOR SYMB	23 FEB 76	22:42:10	51		59	5	0	1	3100
SORTNG	RELOCATABLE	23 FEB 76	22:42:17	52	2	67				3159
TARGET	FOR SYMB	23 FEB 76	22:43:18	53		169	5	0	1	3228
TARGET	RELOCATABLE	23 FEB 76	22:43:35	54	2	158				3397
TRANSF	FOR SYMB	23 FEB 76	22:43:37	55		20	5	0	1	3557
TRANSF	RELOCATABLE	23 FEB 76	22:43:39	56	2	20				3577
FRTAPE	FOR SYMB	04 MAR 76	21:25:25	57		71	5	0	1	3599
FRTAPE	RELOCATABLE	04 MAR 76	21:25:34	58	5	84				3670
SP	MAP SYMB	04 MAR 76	21:25:34	59		1	5	0	1	3759
SP	ABSOLUTE	04 MAR 76	21:26:01	60		865				3760
DP	MAP SYMB	04 MAR 76	21:26:04	61		1	5	0	1	4625
DP	ABSOLUTE	04 MAR 76	21:26:50	62		910				4626

NEXT AVAILABLE LOCATION= 5536

ASSEMBLER PROCEDURE TABLE EMPTY

COBOL PROCEDURE TABLE EMPTY

FORTRAN PROCEDURE TABLE

D NAME	LOCATION	LINK	D NAME	LOCATION	LINK	D NAME	LOCATION	LINK
VIM	66666	26	GEOM	67074	28			