AD

AMSAA TM 98

AMSAA

TECHNICAL MEMORANDUM NO. 98

THE FIRST CONFERENCE OF USERS OF THE MAGIC AND SAM-C PROGRAMS

R. A. Marking

May 1971

AD 728124

Approved for public release;
distribution unlimited.

U.S. ARMY ARMAMENT RESEARCH AND DEVELOPMENT CENTER
ARMY MATERIEL SYSTEMS ANALYSIS AGENCY
ABERDEEN PROVING GROUND, MARYLAND

BEST AVAILABLE COPY

DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the AMC position unless so specified.

BEST AVAILABLE COPY

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY (Corporate author) US Army Aberdeen Research and Development Center Army Materiel Systems Analysis Agency Aberdeen Proving Ground, Maryland 21005	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

THE FIRST CONFERENCE OF USERS OF THE MAGIC AND SAM-C PROGRAMS

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

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

R. A. Marking

6. REPORT DATE May 1971	7a. TOTAL NO. OF PAGES 153	7b. NO. OF REFS
8a. CONTRACT OR GRANT NO. b. PROJECT NO. RDT&E 1T562601A259 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) AMSAA TM NO. 98 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	

10. DISTRIBUTION STATEMENT

Approved for public release, distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY US Army Materiel Command Washington, D.C. 20315

13. ABSTRACT

This report contains the proceedings of the first conference of the users of the MAGIC and SAM-C computer programs held August 6, 7, and 8, 1969, at the Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Maryland. The purpose of the conference was to determine which subprograms constitute the MAGIC and SAM-C programs, to create program source decks, to create benchmark test problems, and to discuss problems of mutual interest.

Also included are the addresses, in expanded form, of the principal users of the two programs. These addresses included corrections and improvements to the two codes.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

UNCLASSIFIED
Security Classification

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Nuclear fluence, flux, dose, or dose rate analysis						
Radiation transport benchmark problems						
Target description benchmark problems						
MAGIC Conference						
SAM-C Conference						

UNCLASSIFIED
Security Classification

ARMY MATERIEL SYSTEMS ANALYSIS AGENCY

TECHNICAL MEMORANDUM NO. 98

THE FIRST CONFERENCE OF USERS OF THE MAGIC AND SAM-C PROGRAMS

R. A. Marking

MAY 1971

Approved for public release,
distribution unlimited.

RDT&E Project No. 1T562601A259

ABERDEEN PROVING GROUND, MARYLAND

ARMY MATERIEL SYSTEMS ANALYSIS AGENCY

TECHNICAL MEMORANDUM NO. 98

RAMarking/paw
Aberdeen Proving Ground, Md.

THE FIRST CONFERENCE OF USERS OF THE MAGIC AND SAM-C PROGRAMS

ABSTRACT

This report contains the proceedings of the first conference of the users of the MAGIC and SAM-C computer programs held August 6, 7, and 8, 1969, at the Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Maryland. The purpose of the conference was to determine which subprograms constitute the MAGIC and SAM-C programs, to create program source decks, to create benchmark test problems, and to discuss problems of mutual interest.

Also included are the addresses, in expanded form, of the principal users of the two programs. These addresses included corrections and improvements to the two codes.

CONTENTS

Page

ABSTRACT 3

1. INTRODUCTION AND PURPOSE 7
   1.1 Copy of Conference Announcement (Abridged) 7
   1.2 Opening Address 11

2. INTRODUCTION OF PARTICIPANTS 12

3. RECENT ACTIVITY WITH MAGIC AND SAM-C 22
   3.1 Recent Activity with MAGIC at AMSAA 22
      3.1.1 Changes to MAGIC 22
      3.1.2 Proposed Plans for MAGIC 27
      3.1.3 Concluding Remarks 28
      3.1.4 Additional Information 28
   3.2 Recent Activity with SAM-C at NEL 34
      3.2.1 Recent Calculations 34
      3.2.2 Corrections/Improvements 35
      3.2.3 Improvements in Progress 36
      3.2.4 Planned Improvements 36
   3.3 Coordination of MAGIC and SAM-C 36
      3.3.1 Discussion 36
      3.3.2 Program Changes 38

4. DEFINITION OF MAGIC AND SAM-C; CREATION OF SOURCE DECKS . . . 38
   4.1 MAGIC 38
      4.1.1 Standard Version 38
      4.1.2 Ad Hoc Problems 40
   4.2 SAM-C 40
      4.2.1 Standard Version 40
      4.2.2 Ad Hoc Problems 40

5. CREATION OF BENCHMARK PROBLEMS 41
   5.1 MAGIC 41
   5.2 SAM-C 41

APPENDIX A MAGIC/SAM-C CONFERENCE PARTICIPANTS 43

APPENDIX B MAGI REPORT MR 6902 (ABRIDGED). 45

APPENDIX C SAM-C BENCHMARK PROBLEM INPUT 59

APPENDIX D LISTING OF THE AMSAA OCTOBER REVISED STANDARD MAGIC PROGRAM 73

DISTRIBUTION LIST 151

Next page is blank.

THE FIRST CONFERENCE OF USERS OF
THE MAGIC AND SAM-C PROGRAMS

1. INTRODUCTION AND PURPOSE

This report presents the proceedings of the first large-scale conference of the users of the MAGIC and SAM-C computer programs. The conference convened at 0915 on August 6, 1969, in Building 328 at Aberdeen Proving Ground, Maryland. Early in the afternoon of the 7th the conferees divided themselves into two groups: those interested in the MAGIC code and those interested in the SAM-C. The conference closed shortly after noon on Friday the 8th.

The following sections contain abridged versions of the announcement letter and of the opening address.

1.1 Copy of Conference Announcement (Abridged).

"AMXRD-AWF

"SUBJECT: Conference for Users of the MAGIC/SAM-C Computer Programs

"1. References:

a. The MAGIC SAM-C Target Analysis Technique, AMSAA Technical Reports 4, 10, 11, 13, and 14.

b. A Geometric Description Technique Suitable for Computer Analysis of Both the Nuclear and Conventional Vulnerability of Armored Military Vehicles, MAGI-6701 (AD 847576).

c. UNC-SAM-2: A FORTRAN Monte Carlo Program Treating Time-Dependent Neutron and Photon Transport Through Matter, UNC-5157 (AD 647470).

"2. The SAM-C Monte Carlo radiation transport program and the MAGIC conventional projectile ray-tracing program were developed by MAGI (Mathematical Applications Group, Inc) under Contract No. DAAD05-67-C-0041 for this agency (ref. a). MAGIC is the computer code that assembles data along selected rays through a target by employing the MAGI originated Combinatorial Geometry technique (ref. b) which utilizes combinations of certain basic solids such as boxes, wedges, etc., to

describe the target. SAM-C is the UNC-SAM-2 Monte Carlo nuclear radiation transport code (ref. c) with the original geometry routines replaced by those of the Combinatorial Geometry technique. These programs have been disseminated to a number of interested government agencies for their use and will be made available to others upon request.

"3. Recent activity with MAGIC has resulted in several advances including the addition of new "library" solids, additional input checking, and faster input processing. These advances are clearly applicable to SAM-C as well and are thus of general interest to both groups. On the other hand, recent activity with SAM-C indicates that the flux-at-a-point routine is incorrectly coded, versions for different computing systems have non-trivial differences in logic/organization, and the computational procedure for carrying out a complete set of calculations is unacceptably complicated.

"4. As a result of the many questions, comments, and suggestions from the personnel involved in using SAM-C/MAGIC, it is clear that everyone involved can benefit from an informal discussion of problem areas and a concerted attack on areas of mutual interest. It is the purpose of this letter to announce an informal, unclassified conference to define and solve problems in the MAGIC and SAM-C computer codes developed under the auspices of this agency.

"5. It is envisioned that the conference will begin on 6 August 1969 at 0900 hours in the Conference Room of ARDC Building 328 under the chairmanship of Mr. R. A. Marking of AMSAA. At this writing it is envisioned that 2 days will be required with a third day allotted only to provide a buffer period.

"6. The first day will be devoted to introductory, informal discussions of individual problems, solutions, and changes to the various codes plus presentations (on MAGIC by Mr. Larry Bain, Methodology Office, AMSAA, and on SAM-C by Dr. Wayne Coleman, Nuclear Physics Branch, NEL) of information gathered too late for inclusion into the reports of reference a. The item, "Introduction of Participants," is envisioned as an opportunity for each participant to identify which program(s) his agency is using, what problems have been encountered, any solutions that have been created, and what specific problem areas he would like to see addressed during the conference period. About 20 minutes will be allotted for each individual.

"7. The second day is planned for the creation of ad hoc working groups to achieve solutions of the problems defined on the first day. An additional day is scheduled to allow an orderly conclusion of the working group projects and the conference as a whole if necessary; the form and content of a conference report will be decided on the last day.

"8. Two items have been of special interest to all those contacted thus far: (1) the establishment of FORTRAN source decks for MAGIC and SAM-C that are not subject to the vagaries of day-to-day changes and are available to be copied by new user agencies or in the event that serious problems develop with an existing source deck; and (2) the creation of "benchmark" test problems that will provide valid tests of all features/options operational using the "archival" or "library" source deck. It is expected that these two items can be introduced in the afternoon of the first day of the conference.

"9. Participation in the conference is encouraged to promote the utility of the SAM-C and MAGIC programs. Representation from your organization is invited. The distribution list is not considered exhaustive and interested personnel within government may be invited to attend by contacting Mr. Marking. Participants are encouraged to bring listings of the current working versions of their program(s).

"10. It is intended that the meeting will result in documented improvements in the SAM-C and MAGIC source programs and/or implementation procedures. This documentation is expected to be in the form of a letter or technical note and copies will be sent to all participants.

"11. The desirability of scheduling similar meetings at regular intervals will be discussed as a means of maintaining lines of communications between MAGIC-SAM-C users."

12. (Administrative)

13. (Administrative)

FOR THE DIRECTOR:

/s/ Morgan G. Smith
MORGAN G. SMITH
Chief, Ground Warfare Division

1 Incl
as
(CF)

PROPOSED CONFERENCE AGENDA

DAY 1

"Introduction and Purpose

Introduction of Participants

Recent AMSAA Activity with MAGIC

Recent NEL Activity with SAM-C

Definition of MAGIC and SAM-C Capabilities and

Creation of Appropriate Source Decks

Creation of Benchmark Problems

DAY 2

Conclusion of the Creation of Archival Source Decks and Benchmark Problems

Formation of Sub-groups to Solve the Problems Defined on Day 1

Discussion and Drafting of Documentation

Consideration of Future Meetings

DAY 3 (If Necessary)

Conclusion of Unfinished Business"

Incl 1 to ltr

1.2 Opening Address.

"Welcome to the 1st Conference of Users of the MAGIC and SAM-C Programs.

"As stated in the Conference letter-announcement, this is to be an informal, unclassified conference of the users of the MAGIC and/or SAM-C computer programs (as well as any of the peripheral programs that might be of mutual interest).

"The purpose of this conference is three-fold:

- to find out where we stand with respect to the actual operation of both SAM-C and MAGIC
  - who is using what
  - on what machines are the codes operating
  - what changes of substance (e.g., packing into 36-bit v. 48-bit words) have been made
  - what kinds of problem areas are being considered (e.g., deep penetration as opposed to close-in transport problems, penetrator fragmentation, x-ray or thermal neutron problems, etc.)
  - what sort of functional and/or theoretical problems are being encountered in operating these codes (e.g., functional identifiers of "0", incorrect evaluation of the uncollided flux, improper coding of SAM-C for inelastic scattering, etc.)
- to define what SAM-C and MAGIC should be capable of
  - there are a number of versions of SAM-C several of which are different enough to require separate operating manuals
  - changes have significantly increased the speed of MAGIC thus making here-to-fore impractical applications worth considering
  - should SAM-C be modified along the lines of UNC-SAM-3 (ENDF/B cross-sections and a non-common energy mesh)
  - should MAGIC calculate vulnerable areas
  - should the geometry processing routines (e.g., GENI, RPPIN, ALBERT, etc.) be called MAGIC and the "driving" or controlling routines such as VOLUM and GRID be handled as separate "packages"
  - should MAGIC employ packing and what effect would its absence have on SAM-C

- to create benchmark problems
  - compatibility of geometric input between MAGIC and SAM-C
  - should they be mathematical tests of all the options or tests of physical acceptability of some combination of both...."

2. INTRODUCTION OF PARTICIPANTS

At the beginning of the Conference the individual participants indicated (1) which program they were using, (2) on which computers was it routinely used (or intended to be used), (3) what were the nature and complexity of the problems treated, (4) what program innovations had been made, and (5) what problems or errors had been encountered. Additionally, it was requested that participants indicate specific program problems to discuss during the Conference.

To accomplish these introductions in an orderly manner, a form covering all of the obvious points of interest was used. Since some of the agencies were represented by more than one person, representatives from the same group using the same program caucused to present a unified picture of their work and problems.

The participants' introductions follow the list of agencies; they are in alphabetical order.

List of Agencies Represented

Aeronautical Systems Division
(Wright-Patterson)

Air Force Armament Laboratory
(Eglin)

Air Force Weapons Laboratory
(Kirtland)

Ground Warfare Division (AMSAA)
(Aberdeen)

Methodology Office (AMSAA)
(Aberdeen)

Naval Weapons Laboratory*
(Dahlgren)

Nuclear Effects Laboratory (BRL)*
(Edgewood)

*No Participant Introduction Form available.

List of Agencies Represented (Cont'd)

SMUPA-DW6*
(Picatinny)

SMUPA-SS
(Picatinny)

SMUPA-TW3
(Picatinny)

SMUPA-VC1
(Picatinny)

RSIC (ORNL)
(Oak Ridge)

Signature & Propagation Laboratory**
(Aberdeen)

Terminal Ballistics Laboratory (BRL)*
(Aberdeen)

Vulnerability Laboratory (BRL)
(Aberdeen)

---

*No Participant Introduction Form available.

**Observer only, no Participant Introduction Form included.

PARTICIPANT INTRODUCTION FORM

Agency: USAF, ASD (ASBRS), WPAFB, Ohio 45433

Name(s) of Representative(s): Gerald Bennett (ASBRS)
Roy Hilbrand (ASVCP)

Program Used: MAGIC

Purpose: To provide target descriptions for use in aircraft vulnerability analyses.

Computer(s) Used: Name IBM Direct Coupled 7044/7094 Word Size 36 bits

Memory Size: Total 32768 Available Unk

Tape Drives: No. 16 No. of Channels 4 1401-Compatible? Yes

Program Requirements: Storage 28K Packed Word Size 35 bits

Tape Drives 2 Links? Yes, 1

Dependence on Assembly Language None

Planned Program Usage: To generate target description for aircraft vulnerable area computation.

Planned Program Changes: Addition of plotting, presented area, and volume subroutines; modifications as required to generate and store data for efficient processing in vulnerable area computation program; further simplifications, as possible, to input descriptive data.

Program Innovations: The use of any body as a target volume subdivision (i.e., as an RPP); the streamlining of MAGIC by stripping out about 16 of the subroutines and recoding of others; restructuring of the Master-Aster array (M-A) deleting some items from the M-A array; recoding and repacking for 36 bit words, viz., 35 bits and one sign bit; changed grid cell generation; changed printout; allowing the attack plane to be outside of the enclosing volume; and disposal of random number generator requirement.

Program Problems/Errors: Core storage (too large); complexity in preparation of data. (Various program errors have been corrected and the corrected listings have been forwarded to AMSAA.)

Program Changes of Immediate Interest: Addition of an airfoil shape to the solid library; a more extensive ARB of perhaps 10 to 12 sides; introduction of "canned" standard aircraft component descriptions (e.g., a pilot).

PARTICIPANT INTRODUCTION FORM

Agency: Air Force Armament Laboratory

Name(s) of Representative(s): Sue Gibson

Program Used: MAGIC

Purpose: To be used with a vulnerable area program to produce vulnerable area program to produce vulnerable areas of foreign targets.

Computer(s) Used: Name CDC 6600 Word Size 60 bits

Memory Size: Total 100,000 Available 32,000

Tape Drives: No. 16 No. of Channels Unk 1401-Compatible? No

Program Requirements: Storage 32,000 Packed Word Size 36 bits

Tape Drives 7 Links? Yes, 4

Dependence on Assembly Language None

Planned Program Usage: Describe foreign air and ground targets in terms of line of sight data.

Planned Program Changes: Complete conversion from IBM 7094 to CDC 6600. Omit TESTG and other unnecessary subroutines to allow room for addition of new figure subroutines.

Program Innovations: Point Burst subroutine has been added and is being checked.

Program Problems/Errors: Lack of storage due to amount of storage allowed to each user, not to the total amount of storage in the CDC 6600.

Program Changes of Immediate Interest: Addition of new figures and reduction of amount of storage required.

PARTICIPANT INTRODUCTION FORM

Agency: AFWL, Kirtland AFB, New Mexico 87117

Name(s) of Representative(s): Michael J. Paul - AFWL (WLRAS)
A. Kris Widdison - AFWL (WLCP-M)

Program Used: SAM-C

Purpose: Both deep-penetration (in air) and close-in transport (concrete) problems, primarily neutrons, but including prompt and secondary gammas.

Computer(s) Used: Name CDC 6600 Word Size 60 bits

Memory Size: Total 300K (w/o extended core) Available $1M_8$ (w/extended core) $\cdot 325K_8$ (w/o extended core)

Tape Drives: No. 10 No. of Channels 9 1401-Compatible? Unk

Program Requirements: Storage generally 120K Packed Word Size 45 or 60 bits
Tape Drives 1-3 Links? None
Dependence on Assembly Language some, but easily changed

Planned Program Usage: Hard-rock silo configurations and state-of-art neutron and gamma transport problems.

Planned Program Changes: Complete revision of input to be more understandable and logical and easier to punch. Combined time-energy-angular dependent source input (allowing input of flux from a preliminary discrete ordinates code).

Program Innovations: Free-form reading routine (eliminates need for formatting input). Cut down flux printing by 50% by eliminating extraneous lines (e.g., ΔE). Eliminate need to change NXS = ... and NDQ.... cards (e.g., add a parameter, say ENDM, to end of master array (COMMON DUM (250), MASTER (30000), ENDM), then NDQ = LOCF (ENDM) - LOCF (Master)).

Program Problems/Errors: None.

Program Changes of Immediate Interest: ENDF/B cross sections, inelastic scattering improvements; graphic geometry display; better geometry checking.

PARTICIPANT INTRODUCTION FORM

Agency: AMSAA, Ground Warfare Division (GWD); Methodology Office (MO)
Aberdeen Proving Ground, Maryland 21005

Name(s) of Representative(s): L. Bain (MO)
R. Lake (GWD)
J. Brewer (GWD)

Program Used: MAGIC

Purpose: Conventional vulnerability by 4" cells and/or areas for combat vehicles and aircraft.

Computer(s) Used: Name BRLESC I & II Word Size 68 bits

Memory Size: Total 96K Available 48K

Tape Drives: No. 8 No. of Channels 4 1401-Compatible? Yes

Program Requirements: Storage 48 Packed Word Size 30

Tape Drives 4 Links? No

Dependence on Assembly Language Yes but easily avoided

Planned Program Usage: Conventional vulnerability of combat vehicles and aircraft (both rotary and fixed wing types).

Planned Program Changes:

1) Thirty bit packing for triplets and scalars.
2) Shielded areas.

Program Innovations: Rewrite program flow to minimize presence of unnecessary steps, extraneous comments, and blank cards.

Program Problems/Errors: None.

Program Changes of Immediate Interest: No genuinely pressing problems.

PARTICIPANT INTRODUCTION FORM

Agency: Picatinny Arsenal, Dover, N. J.

Name(s) or Representative(s): Robert Kesselman - VC-1
John Saarmann - VC-1
Robert Barnas - SS
John Burgio - TW3

Program Used: SAM-C NEL Version

Purpose: To obtain running version on IBM 360 for radiation transport and shielding calculations.

Computer(s) Used: Name IBM 360 Word Size 32/64 bits

Memory Size: Total Unk Available 200K

Tape Drives: No. 8 No. of Channels 9 1401-Compatible? Yes

Program Requirements: Storage 200K Packed Word Size 64 bits
Tape Drives 3 Links? Unk
Dependence on Assembly Language one subroutine

Planned Program Usage: Transport and Shielding

Planned Program Changes: In January 1970 Picatinny Arsenal will start using CDC 6500; therefore, the debugging effort on the conversion has been suspended.

Program Innovations: (See comment above.)

Program Problems/Errors: (See comment above.)

Program Changes of Immediate Interest: (See comment above.)

PARTICIPANT INTRODUCTION FORM

(Observer)

Agency: Radiation Shielding Information Center
Oak Ridge National Laboratory
P. O. Box X
Oak Ridge, Tennessee 37830

Name(s) of Representative(s): Robert W. Roussin

Program Used: SAM-C

Purpose: For distribution to anyone who wants the program. (RSIC operations are sponsored by the AEC, DASA, and NASA.)
We have the CDC 6600 version for distribution (but no 6600 at ORNL).

Computer(s) Used: Name Word Size

Memory Size: Available

Tape Drives: No. of Channels 1401-Compatible?

Program Requirements: Storage Packed Word Size
Tape Drives Links?
Dependence on Assembly Language

Planned Program Usage:

Planned Program Changes:

Program Innovations We are Interested in:

1) IBM 360 version.
2) Version with ENDF/B cross sections.

Program Problems/Errors:

Program Changes of Immediate Interest:

PARTICIPANT INTRODUCTION FORM

Agency: BRL, Vulnerability Laboratory, Aberdeen Proving Ground, Md. 21005

Name(s) of Representative(s): M. J. Reisinger

Program Used: MAGIC

Purpose: Currently being used to debug target descriptions for Electronics Command, Army Tank Automotive Command, Missile Command, Munitions Command, Weapons Command, Nuclear Effects Laboratory, Test & Evaluation Command and Falcon Research & Development.

Computer(s) Used: Name BRLESC Word Size 68 bits

Memory Size: Total 120K Available 80K

Tape Drives: No. 3 No. of Channels 7/9 1401-Compatible? Yes

Program Requirements: Storage 48 Packed Word Size 30

Tape Drives 4 Links? No

Dependence on Assembly Language Depends on version

Planned Program Usage:

1) "Graphic Program" being developed from NASA program by L. Bain and M. J. Reisinger.

2) Recognition of heat projectile improper detonation from target description.

3) Point burst program with emphasis on components.

Planned Program Changes: Generalized Ellipsoid (i.e., not restricted to ellipsoids of revolution). Elimination of enter-leave table philosophy in favor of a more direct approach which is intended to reduce tracking time.

Program Innovations: Graphics Package.

Program Problems/Errors: A more detailed description of targets than done in the past (example, M60A1 with approximately 2500 bodies) is rapidly approaching our computer system time and size limit: a 1300 body description is using 64K, will 2500 bodies use less than the available 80K? Computer time on our system forces partial runs for graphics (need about four hours, large memory). Summation: need better computer.

Program Changes of Immediate Interest:

1) Development of support subroutines such as generalized components (wheels, ammunition, engine, etc.) that require location, orientation, and relative size that lead to computer generated bodies (solids).

2) Development of programs that would construct the optimized solid for a body from an input consisting of point data read directly from engineering drawings.

3. RECENT ACTIVITY WITH MAGIC AND SAM-C

The three sections that follow consist of the material presented verbally at the conference plus one or two minor additions or modifications.

3.1 Recent Activity with MAGIC at AMSAA. (Presentation by Larry Bain)

The recent activity with MAGIC at AMSAA falls into one of two categories: program changes or proposed plans. Each category is discussed separately.

3.1.1 Changes to MAGIC. This category is divided into three subtopics:

- Additions.
- Modifications.
- Corrections.

a) Additions. Of primary interest, three new solids have been added:

- TEC (Truncated Elliptic Cone).

  i) Height vector does not need to be perpendicular to the base ellipse.

  ii) Specify (Table 3.1)

  - V - vertex of base ellipse,
  - H - height vector,
  - M - semimajor axis of base ellipse,
  - m - semiminor axis of base ellipse, and
  - R - ratio (base ellipse/top ellipse),

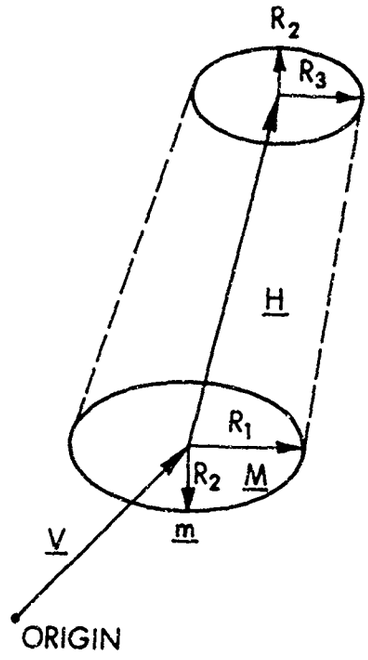
    viz., R = (R1/R3) = (R2/R4).

(The normal height vector is computed in GENI; $\underline{n} = \underline{M} \times \underline{m}$; it is necessary to change the sign of $\underline{n}$ if $\underline{H} \cdot \underline{n} < 0$.)

TABLE 3.1 CARD INPUT FOR THF NEW SOLIDS

Body Type	3-Letter ID 4 - 6	11-20	21-30	31-40	41-50	51-60	61-70	No. of Cards Needed
Truncated Elliptic Cone	TEC	V_x	V_y	V_z	H_x	H_y	H_z	1 of 3
		M_x*	M_y*	M_z*	m_x*	m_y*	m_z*	2 of 3
		R						3 of 3
Torus	TOR	V_x	V_y	V_z	N_x	N_y	N_z	1 of 2
		R1	R2					2 of 2
Arbitrary Curved Surface	ARS							
		M	N					1 of n
		X(1,1)	Y(1,1)	Z(1,1)	X(1,2)	Y(1,2)	Z(1,2)	3 of n
		.						
		.						
		.						
		X(1,N)	Y(1,N)	Z(1,N)				N+2 of n
		X(2,1)						
		.						
		.						
		.						
		X(M,1)						
		.						
		.						
		.						
		X(M,N)	Y(M,N)	Z(M,N)				

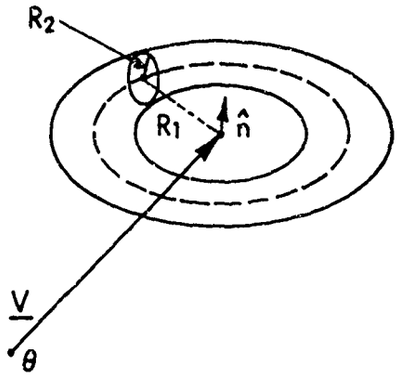
*M → semimajor axis
m → semiminor axis



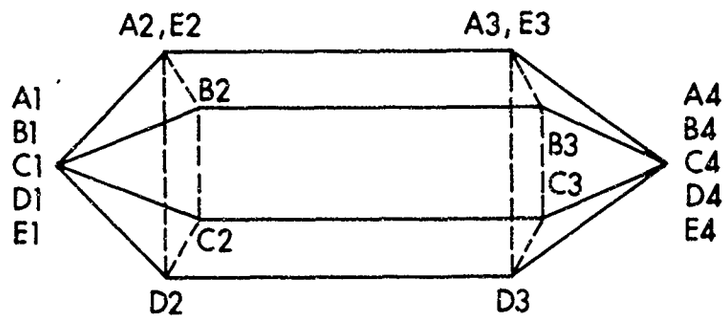
- TOR (Torus)

Specify (Table 3.1)

- V - vertex,
- n - normal vector (normal to the plane bisecting the torus),
- R1 - radius from V to the mid-point of the torus' cross section, and
- R2 - cross sectional radius (R1 $\geq$ R2).



- ARS (Arbitrary Surface)
  - i) Specify M curves of N points (see Table 3.1).
  - ii) The number of words of memory is 92 + M*N*4.
  - iii) An ARS can be described in more than one way.



Example (using the figure above).

One Approach (Solid Lines)			Another Approach (Dashed Lines)		
M = 5	N = 4		M = 4	N = 5	
curve 1	pt	A1, A2, A3, A4	curve 1	pt	A1, B1, C1, D1, E1
curve 2	pt	B1, B2, B3, B4	curve 2	pt	A2, B2, C2, D2, E2
curve 3	pt	C1, C2, C3, C4	curve 3	pt	A3, B3, C3, D3, E3
curve 4	pt	D1, D2, D3, D4	curve 4	pt	A4, B4, C4, D4, E4
curve 5	pt	E1, E2, E3, E4			
Figure closed by duplicate curve			Figure closed by duplicate points		
Note:	pt A1 = B1 = C1 = D1 = E1		pt A2 = E2		
	pt A4 = B4 = C4 = D4 = E4		pt A3 = E3		

The addition of these new solids has required the addition of auxiliary subroutines:

- QRTIC to solve 4th degree equation.
- CUBIC to aid QRTIC.

- CROSS to compute vector cross products.
- DOT to compute dot products.
- UNIT to compute unit vectors.
- ARIN to process the ARS input.

Additional coding is also similarly required in subroutine CALC to compute normals and in GENI, G1, and WOWI.

Quite apart from the new solids, coding has been incorporated in Geni to increase input checking as follows:

- Checks of vector perpendicularity in BOX, RAW, REC, and TEC.
- Checks of TRC radii to ensure that $R_B \neq R_T$.
- Checks of TOR radii to ensure that R1 is not less than R_2 (v.s.).

Schematically, an option has been added to suppress tape 8 (the monitor output) output (except for error messages) when writing tape 1.

Finally, two control subroutines have been created to assess quantities other than line-of-sight thickness:

- AREA to compute presented areas (the ray is traced to its first contact).
- MOMENT to compute moments of inertia (and as a by-product, the center of gravity, total weight, total volume, mean angle of incidence, and the mean cosine of incidence are also computed).

b) Modifications. Seven modifications have been made:

- The ELL input has been optimized.
  - i) Present input (cc 7-10 = 0 on card 1) is both foci plus the length of the major axis.
  - ii) Optimal input (cc 7-10 $\neq$ 0 on card 1) requires the vertex, a vector representing the semimajor axis, and the scalar length of the semiminor axis.
- Computer word packing has been converted from 45 bits/word to 30 bits/word (it is estimated that 30 bit packing runs about 30 percent faster on BRLSEC I and II).

- The solid input section has been revised by eliminating FLOCON and DIGCON and substituting the F-type format specification.
- SENSESWITCH settings have been eliminated and their control data read in on punch-cards.
- The data output coding in TRACK has been revised to eliminate SETUP and ISIGN by using I and F-type format specifications.
- A version of MAGIC has been written in "standard" FORTRAN (incidentally, this version runs slower on the BRLESCs than any of the versions already mentioned).
- The control logic in the main program has been modified so that VOLUM can be run without reading in the Identification Table.

c) Corrections. Four major subroutines were found to have more or less subtle errors:

- VOLUM faltered when G1 tried to combine regions of the same "item" code because VOLUM requires that each region be processed separately; a special exit was added to G1 to correct this condition.
- TESTG suffered a similar fate but to a greater extent since the item data was not loaded into core prior to the execution of TESTG; an additional special exit was added to G1 to rectify this condition.
- GENI computes data for the normal vector to the base ellipse in the TEC but failed to ensure that the normal was an inward rather than an outward normal; a check was added so that if $\underline{H} \cdot \underline{n}$ is negative the direction of $\underline{n}$ is reversed.
- CALC was unable to correctly calculate normal thickness through adjacent regions with the same item code (cf., VOLUM and TESTG); a modification to the existing exit in G1 to compare the item code of the next region with the item code of the previous region was made to allow continuing the normal ray.

3.1.2 Proposed Plans for MAGIC. Activity in four areas is being carried on:

- Compatibility with SAM-C - let MAGIC do some of the geometry processing for SAM-C.
- Eliminate part or all computer word packing.

- Couple the target description ray tracing directly to outputs such as vulnerable area, kill probabilities, etc.
- Addition of a graphics capability via plotters, line-printers, CRT displays, or all three.

3.1.3 Concluding Remarks. The actual changes required to implement the three new solids are discussed in the MAGI report, A Description of Three Additional Bodies for the MAGIC Conventional Vulnerability Program, by J. R. Davis and M. Moskowitz (MAGI report MR 6902, May 1969). An abridged version of this report consisting of the main portion of the text appears as Appendix B.

In any direct dealings with the coding of the MAGIC program, a knowledge of the core storage layout, input data requirements, etc., is essential. Figures 3.1 through 3.4 supply the requisite data:

- Figure 3.1 is a map of the MASTER/ASTER array showing storage for the processed geometry data. Both 45-bit and 30-bit packing versions are shown. Variable names beginning with L are the locations of each set of data in the MASTER/ASTER array.
- Figure 3.2 concludes the map of the MASTER/ASTER array showing storage of the identification table and the "working" storage used at run time.
- Figure 3.3 is a map of the pointers to the location of the solid data.
- Figure 3.4 is the map of the storage for the arbitrary surface (ARS).

Finally, to appreciate the relationships between various routines, Figure 3.5 displays the many auxiliary routines of MAGIC and their relationship to the main of "driver" routines.

3.1.4 Additional Information. In October, 1969, several runs were made using the AMSAA "Revised Standard MAGIC" (Appendix D) on a number of different computing systems. The geometry input consisted of a description which we shall call the "December '68 Master Target." This target is comprised of 701 solids and 904 regions; none of the three new solids were used. About 5 man-months were required to create the description. The following driver routines and their input were used:

GRID	0° Az 0° Elev	1015 cells
VOLUM	Head-on	1015 cells
AREA	Head-on	4189 cells
TESTG	------	2 rays

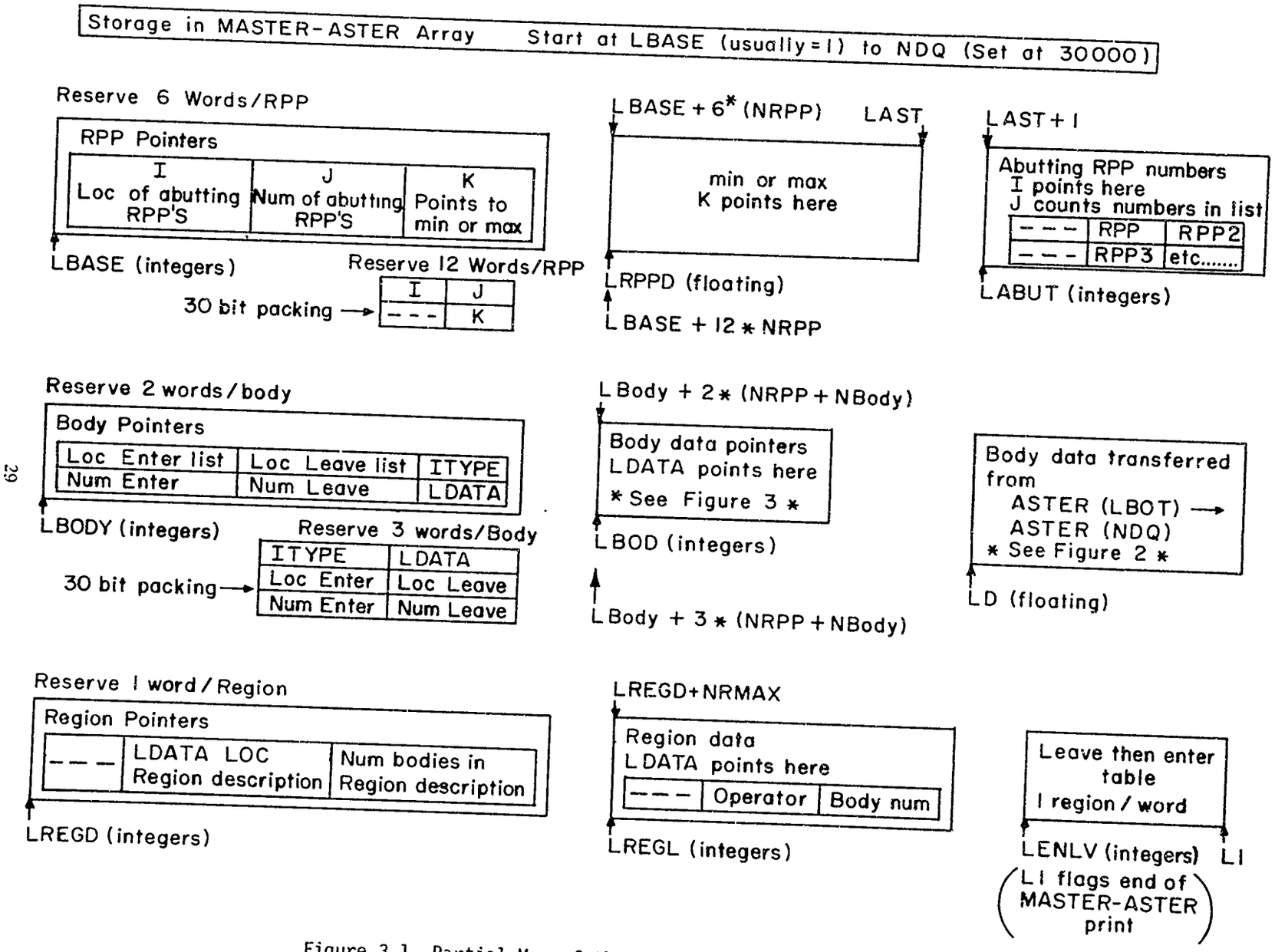


Figure 3.1 Partial Map of the MASTER-ASTER Array

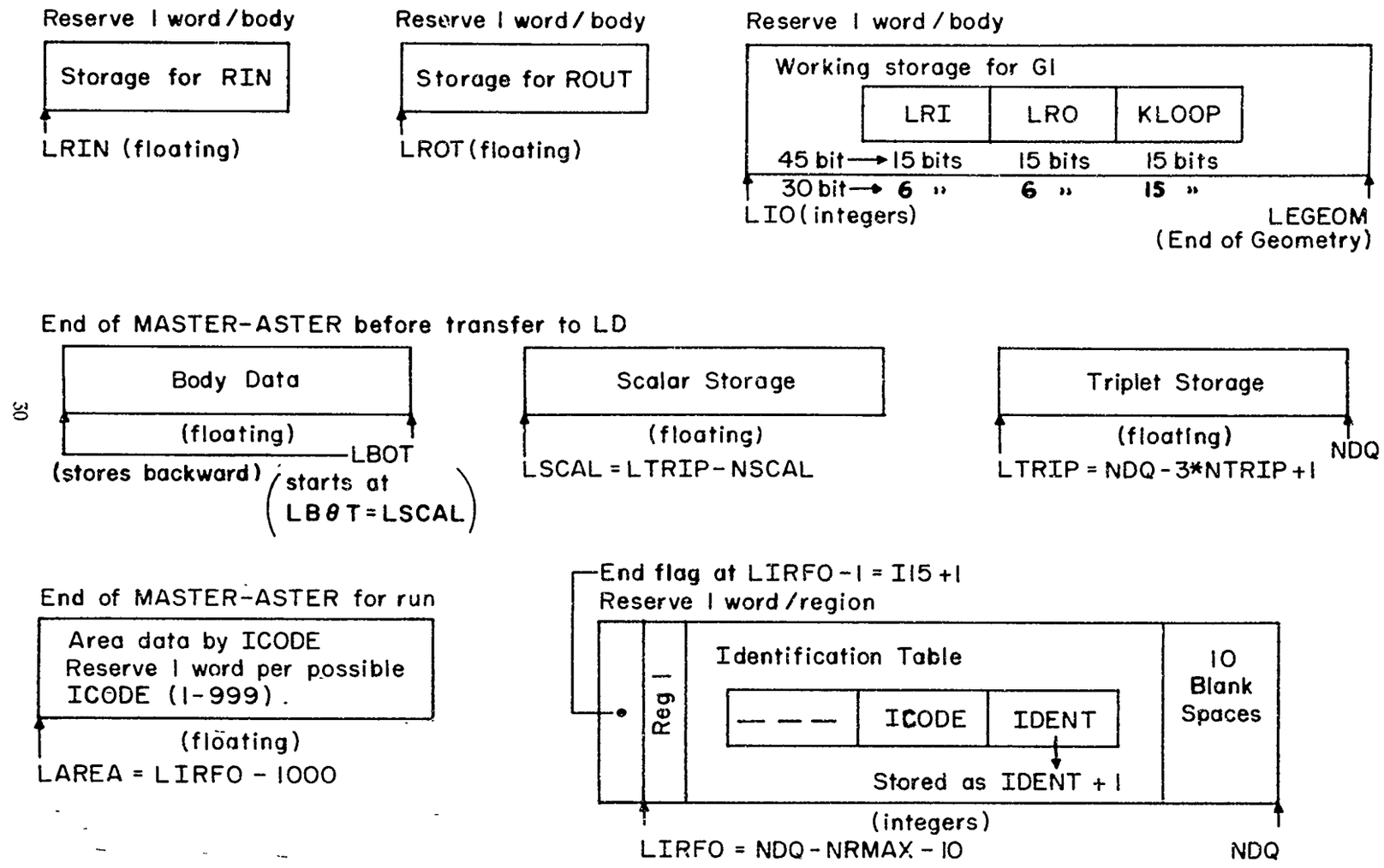


Figure 3.2 Map of the Remainder of the MASTER-ASTER Array

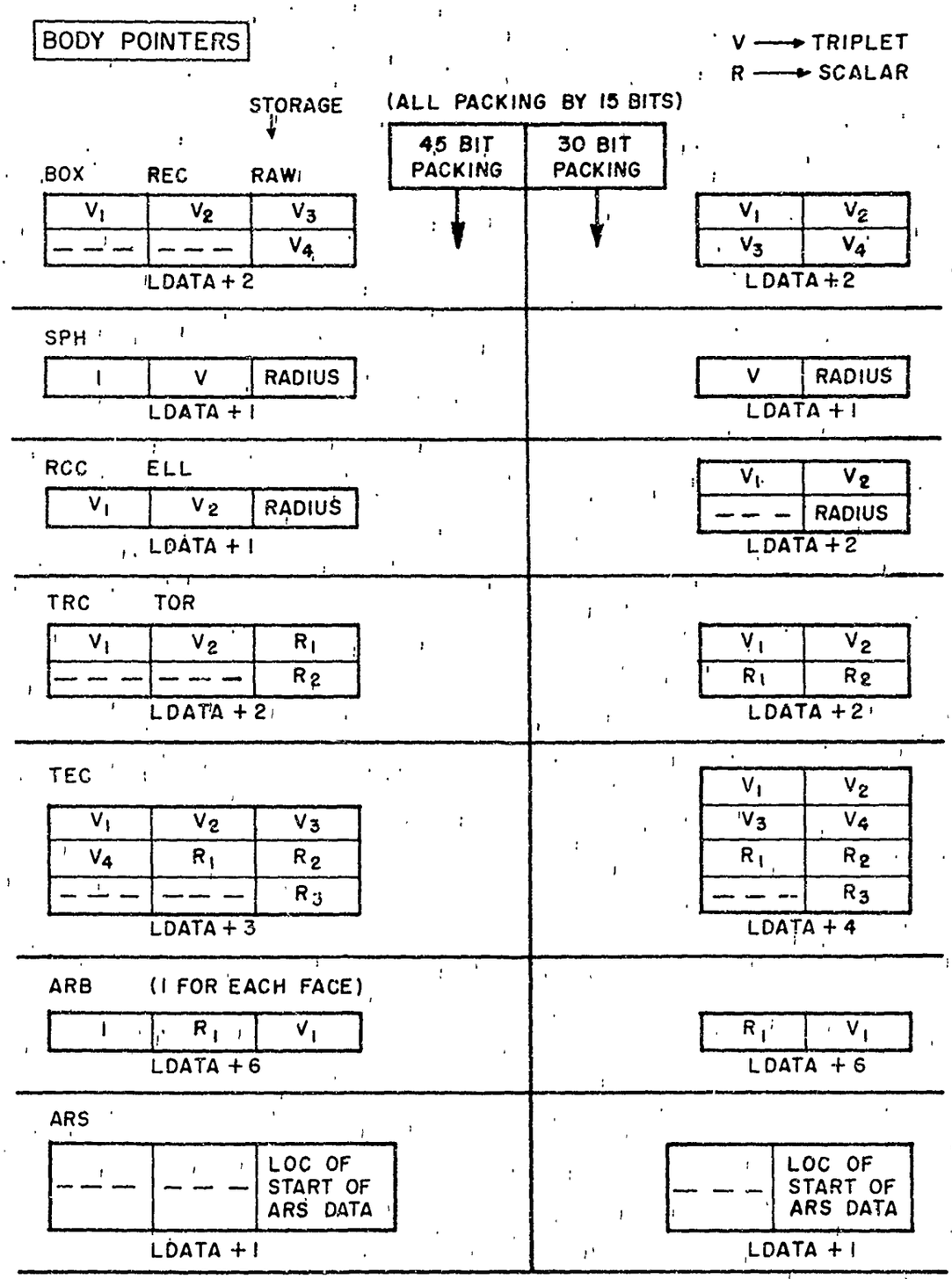


Figure 3.3 Map of Pointers to Solid Data

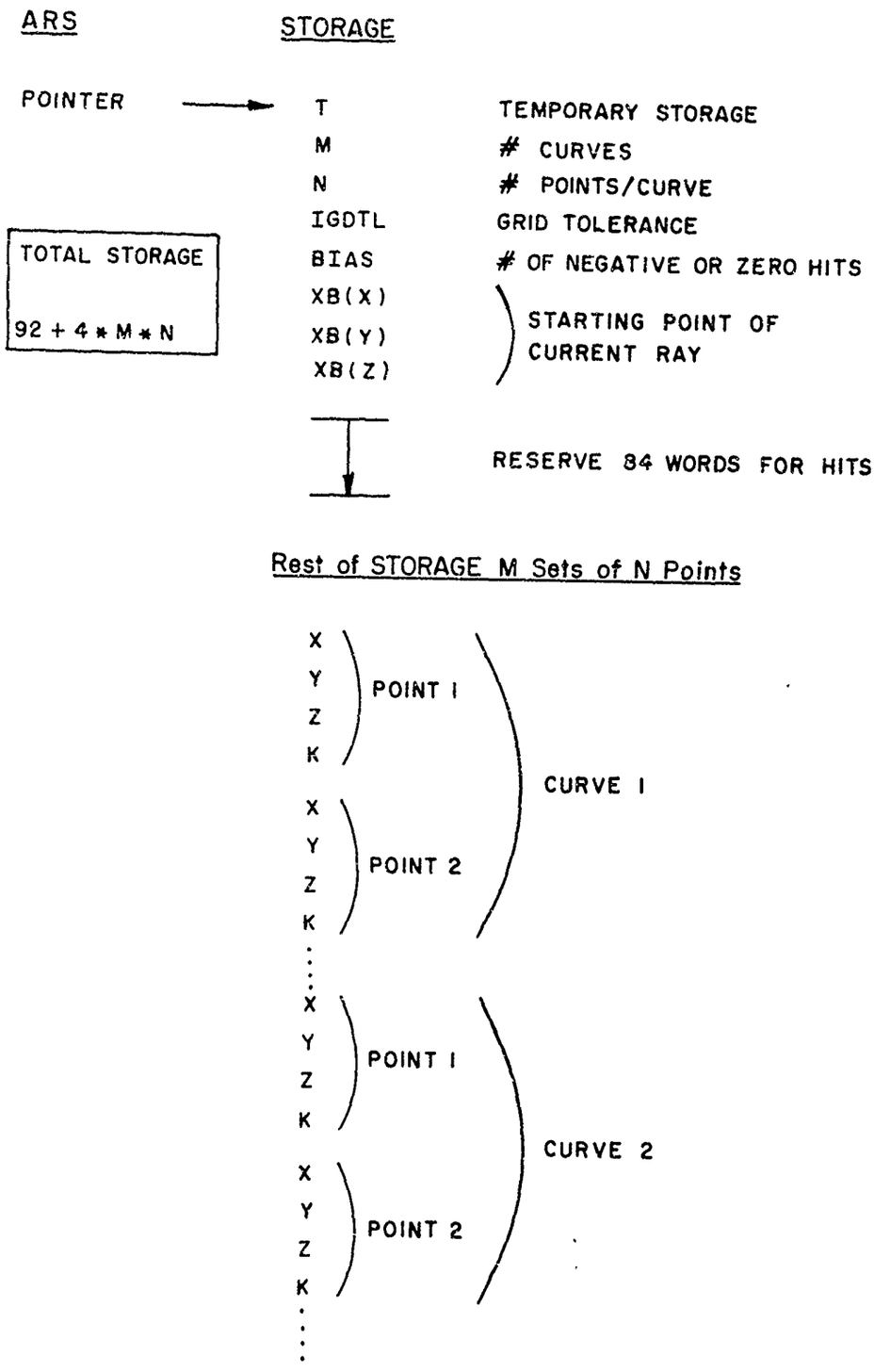


Figure 3.4 Storage for the ARS

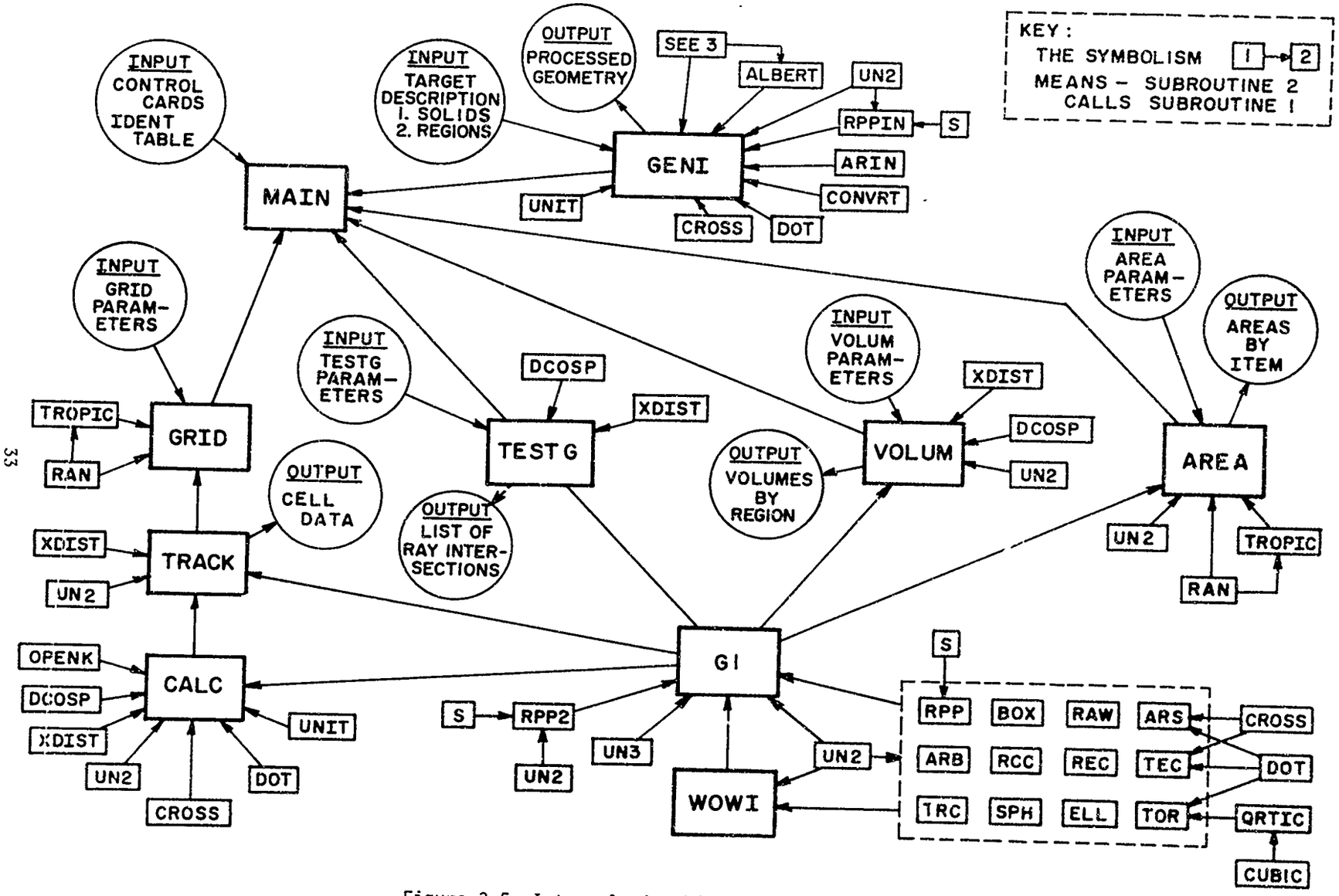


Figure 3.5 Interrelationships of MAGIC Routines

Table 3.2 presents the results of the various computer runs.

TABLE 3.2 COMPUTER SYSTEM TIME REQUIREMENTS

System	Location	Time (in minutes)	Ratio
BRLESC I	ARDC, APG, Md	133	1
BRLESC II	ARDC, APG, Md	55*	.41
CDC 6600	New York Univ	15	.11

*Using the on-line printing capability instead of "dumping" onto tape for off-line tabulation.

3.2 Recent Activity with SAM-C at NEL. (Presentation by Wayne Coleman)

The recent activity with SAM-C at NEL falls into one of three categories: recent calculations completed using SAM-C, completed corrections and improvements to the SAM-C code, and plans for the future.

3.2.1 Recent Calculations. This category is divided into three subtopics:

- Calculation of the energy dependent gamma flux at 3 feet above an infinite Co-60 "fallout" field.
- Calculation of the energy dependent and total neutron fluence at various positions within the Ralph J. Truex (Tandem Van de Graff) accelerator at the Nuclear Effects Laboratory (Edgewood Arsenal).
- Calculation of the energy dependent neutron fluence in an environment simulating that used in Operation HENRE.

a) Infinite Fallout. Although these calculations did not constitute a comprehensive test of the geometric capabilities of SAM-C, very good agreement with the known solution of this problem was obtained.

b) Accelerator. These calculations include the most complex geometries that have been simulated to date at NEL and were included for that reason. The physical results unfortunately cannot be compared directly to any results of experiments or any other calculations.

c) HENRE. This problem demonstrates how, under certain restrictions, ad hoc changes can be made to produce results for source angular distributions other than 4π-isotropic or monodirectional. The results compared favorably with unfolded flux spectra from experimental activation measurements when the SAM-C results were used as an "input guess spectrum" in the unfolding calculations.

3.2.2 Corrections/Improvements. Corrections are indicated by "C" while changes that are more in the category of improvements are indicated by "CI." C/CI's are by order of their appearance in the program.

a) Variable Dimensioning (CI). On machines that assign priority based on memory requirements, adjustable sizes of blank common and labeled common CROSA are an operational necessity. Variable dimensioning in SAM-C is accomplished by creating an artificial "main" program in which dimensioning is accomplished.

b) SEEK (C). Change E = 1 to I = 1 in TUNC and MONTE.

c) SOUCAL (C). Set "KKMAX = K-1" between FORTRAN statement numbers (S.N.) 780 and 925.

d) CARLO (C). The calculation of flux-at-a-point following an inelastic collision is incorrect. See the August RSIC Newsletter (No. 57) for details.

e) SOUPIC (C/CI). Volume source additions may be made by modifying the coding shortly after FORTRAN S.N. 800.

f) ARB (C). T was inadvertently used as a variable name to represent time in a labeled common and temporary storage in the routine (S.N. 50 and 50 + 1).

g) FAP (CI). The coding for identifying point flux contributions by region appears here.

h) FAP (C). To correctly calculate flux-at-a-point add COMMON/LSU/LSURF.

i) GE (C). (1) The scattering problem look-up for neutrons was referencing the wrong area of common for neutron anisotropic scattering; cf., the RSIC version. (2) The third argument in the call to SEEK (S.N. 185) should be 10 and not 11. (3) Format 106 ends incorrectly - replace "10" with any desired E- or F-type format.

j) SUBED (C). Change "NRMAX" to "NDET" at S.N. 121.

k) SOUCAL (C). Change Format 402 (not all versions). (CI) finally changes to compute statistics on the total flux or fluence for flux-at-a-point would be desirable.

3.2.3 Improvements in Progress. Modifications to calculate flux at a point as a function of angle or direction as well as energy are well underway at NEL.

3.2.4 Planned Improvements. Neutron cross section representation is earmarked for extensive study. Key inputs or approaches to this subject include UNC-SAM-3, an ENDF/B cross-section processor, and a thorough examination of the representation of neutron elastic, angular scattering distributions. Total, elastic, and non-elastic cross section data are also expected to be involved.

3.3 Coordination of MAGIC and SAM-C.

Both MAGIC and SAM-C process card-image target description data and store the results in the MASTER/ASTER array. Both can compute volumes. If the conventional component/space code table has been created, a trivial program (Program COMPAS) exists to convert this data to a region/chemical composition assignment table. The following two sections describe the changes required so that target descriptions can be utilized by either MAGIC or SAM on a wide variety of computational equipment with a minimum of difficulty. The goal, of course, is to permit the creation of a library of target descriptions which can serve two purposes: they can reduce duplication of effort and they can lend insight into what a given agency considers an adequate degree of descriptive detail.

3.3.1 Discussion. This category is divided into two subtopics: a description of the proposed tape's contents and a discussion of how the tape should be created.

a) Description of the Tape Contents. The first step is to identify the data available for the library tape; these data are displayed in Table 3.3.

TABLE 3.3 TARGET DESCRIPTION DATA AVAILABLE

MAGIC	SAM-C
"Processed" (GENI) Target	Output Data (MASTER-ASTER*)
Region Identification Table	Composition Assignment Table
Volumes (Optional)	Chemical Composition Definitions
Region Weights (Optional)	"Processed" (BAND/BEDIT) Cross Section Data (MASTER-ASTER)
Moment of Inertia (Optional)	Volumes (ASTER)
Target Description Title	
"Raw" (Card-Image) Target Description Solid & Region Data	

*The SAM-C program stores these data in a location different from that used by MAGIC.

TABLE 3.4 LIBRARY TAPE CONTENTS: PARTS 1 AND 2

"Block" 1. Identification (80A1)
Target Description Title

"Block" 2. Table of Contents

(1) Solid/Region Table Format (cc 1-5)

0 : GENI Input form

$\neq 0$: GENI Output form

For (2) through (8), 0 means the category was not included while a non-zero entry indicates that the data was included

(2) Region Identification Table (cc 6 - 10)

(3) Composition Assignment Table (cc 11 - 15)

(4) Volumes (cc 16 - 20)

(5) Chemical Composition Data (cc 21 - 25)

(6) Region Total Weights (cc 26 - 30)

(7) "Organized" Cross-Section Data (cc 31 - 35)

(8) Moment of Inertia Data (cc 36 - 40)

(9)-(10) (Spares) (cc 41 - 50)

cc 51-52 Unit Systems (Examples: IP for inches, pounds (& seconds), CG for centimeters, grams, seconds, etc.)

cc 53 Coordinate System "Handedness" L = left, R = right

cc 54-80 Location of Geometric Origin with respect to the Reference Origin (e.g., for "tanks" the reference origin is frequently the intersection of the turret datum line and the vehicle centerline) (3E9.2)

The second step is to determine the data to be loaded and in what order. Clearly, the target description title plus some type of flagging to indicate the data categories available should appear near the start of the tape to minimize the time spent in identifying the tape contents. It is therefore proposed that the first tape "block" consist of ten flag words as described in Table 3.4. Beyond these two points, any data that describe the target are suitable for inclusion on the tape.

b) Approach. It was the consensus of the program users that, while all library tapes should be in BCD format for transmittal, the basic target description solid and region data should be in card-image rather than in "processed" (GENI output) form; except for a few installations, the computer time spent in reprocessing the "raw" data is negligible compared with the time that would be used to convert the data into the form required by those of a different installation. It was also agreed that the raw data approach would place the least number of restrictions on the internal operating procedures of any given MAGIC or SAM-C program user.

3.3.2 Program Changes. Some coding changes of a very minor nature will be required. The routines affected are:

a) MAGIC. The main program plus GENI, VOLUM, the moment of inertia, and the vulnerable area routines will require modification.

b) SAM-C. TUNC, GENI, DR, and VOLUM are involved. In SAM-C, VOLUM will generally only require an additional output statement.

4. DEFINITION OF MAGIC AND SAM-C; CREATION OF SOURCE DECKS

This provided the first opportunity of the conference for the conferees to determine the course of action to be followed. The two sections that follow present the consensus achieved by the conferees in defining what the capabilities of the two programs should be and in determining how these capabilities should be achieved.

4.1 MAGIC.

The following sections represent the major areas discussed at the conference together with the results of these discussions.

4.1.1 Standard Version. A consensus was achieved in four areas:

a) Input. The input to MAGIC will consist of

- RPP Data,
- Solid Description Data, and
- Region Description Data.

b) Subprograms. The subprograms are characterized as "geometry processing" (including some testing):

- GENI,
- ALBERT,
- RPPIN, and
- ARIN,

or "ray tracking" (but not in the sense of GRID which is considered a "driver routine"):

- G1,
- WOWI/RPP2,
- RPP/body routines + TOR, ARS & TEC, and
- Auxiliary body routines such as QRTIC, UNIT, etc.

c) Program Features.

- No packing.
- It should be possible to use an RPP to subdivide the target, itself.
- Drop TESTG.
- Drop FLOCON and DIGCON.
- Output the processed geometry which should consist of titling data, the geometry data, and the functional identification table.

d) Tests. It was agreed that TESTG as a random but supposedly complete test of the description is inefficient and should be dropped; in its place, a driver routine of interest - such as GRID - should be used. Input testing was considered valid and at least the following tests should be available:

- Legitimacy of solid names.
- Vector perpendicularity for boxes, RAW's, and the REC.
- Equal radii in the TRC.
- Region checking (on an optional basis).
- "4-points-in-a-plane" in the ARB.
- Degenerate plane in the ARB.

- Proper ordering of RPP input.
- Proper structure creation by contiguous RPP's (since the structure that would enclose all RPP's must be in the shape of an RPP).

4.1.2 Ad Hoc Problems. Although there was not enough time for the formal formation of working groups to solve ad hoc problems, the following problems were defined for solution by any interested groups or individuals.

- The creation of a technique for the arbitrary designation of solids to have the special characteristics currently displayed by the RPP.
- The identification of methodological differences in the routines that form MAGIC between versions held by the several agencies using MAGIC.
- The establishment of a methodology for creating "library" routines (for such configurations as wheels, people, etc.) which can be processed as a unit rather than a set of distinct subsolids.
- The establishment of a uniform system of flags for transmittal of the processed geometry.

4.2 SAM-C.

SAM-C was not the subject of serious discussion until late in the afternoon of the second day. Because the SAM-C program is so large and complex, our attitude toward it is considerably different than our attitude toward MAGIC. In the first place (and of overriding importance), there are few computing facilities capable of efficiently executing the SAM-C programs; secondly, a substantial amount of understanding of the code and the manner in which it attempts to solve problems, and of the problems themselves, is required to achieve any sort of successful solution. Keeping these points in mind, the following sections present the consensus reached.

4.2.1 Standard Version. For the time being at least, the standard version of SAM-C will be that obtainable from RSIC at Oak Ridge. In brief, this version features Combinatorial Geometry inputs identical to those of Standard MAGIC with one important difference: triplet and scalar inputs will be allowed.

4.2.2 Ad Hoc Problems. Although time was again insufficient, the following study/problem areas were defined:

- The abolishment of computer word packing to the greatest practical extent.

- The incorporation of ENDF/B cross sections - preferably using a noncommon energy mesh.
- Bonafide time dependence: time dependence currently assumes the source is a separable function of time.
- Simplification of input preparation.

5. CREATION OF BENCHMARK PROBLEMS

Although time constraints precluded the actual creation of test problem input, it was possible to indicate what the benchmark problems should include.

5.1 MAGIC.

The test problem input should meet five conditions:

- Use only the solids generally available; place the three new solids at the end of the solid table for possible deletion.
- Use all three region operator symbols (+, -, OR).
- Use 1 RPP to enclose the target (Wright-Patterson AFB will use a BOX).
- Employ solids in such a way that they present overlaps and contiguities.
- Similarly, ensure that at least one situation arises where more than one following region has the same functional identification code as the region in front (to ensure that the correct normal thickness is being computed).

In addition, it is desirable to introduce a few deliberate errors to ensure that internal error checks are operative.

5.2 SAM-C.

It was decided that more than one benchmark problem should be created to enable checking the modeling of the physical solution and the execution of the code. Two benchmark problems were agreed upon:

- The "infinite" fallout problem (RSIC Benchmark Problem No. 4), and
- AFWL "Rocket" Geometry.

Physical/code input for these problems is presented in Appendix C.

Next page is blank.

APPENDIX A. MAGIC/SAM-C CONFERENCE PARTICIPANTS

	Name	Agency	City, State
1.	Robert E. Barnas	Picatinny Arsenal	Dover, New Jersey
2.	Sue Gibson	AFATL	Eglin AFB, Florida
3.	Larry Bain	AMSAA (Methodology)	APG, Md.
4.	Matthew J. Reisinger	BRL (VL)	APG, Md.
5.	John A. Zook	AMSAA (ASA)	APG, Md.
6.	Thomas Jankunis	Picatinny Arsenal	Dover, New Jersey
7.	Jesse W. Brewer	AMSAA (ASA)	APG, Md.
8.	Donald W. Mowrer	BRL (VL)	APG, Md.
9.	Robert E. Walther	BRL (VL)	APG, Md.
10.	Lewis G. Gotze	BRL (VL)	APG, Md.
11.	John Saarmann	Picatinny Arsenal	Dover, New Jersey
12.	Michael J. Paul	AFWL (WLRAS)	Albuquerque, N. M.
13.	A. Kris Widdison	AFWL (WLDC-PD)	Albuquerque, N. M.
14.	Kenneth Richer	BRL (S&PL)	APG, Md.
15.	George H. Connor, Jr.	BRL (NEL)	Edgewood Arsenal, Md.
16.	Robert W. Roussin	ORNL - RSIC	Oak Ridge, Tennessee
17.	Wayne A. Coleman	BRL (NEL)	Edgewood Arsenal, Md.
18.	W. B. Beverly	BRL (NEL)	Edgewood Arsenal, Md.
19.	Janet Lacetera	BRL (NEL)	Edgewood Arsenal, Md.
20.	Richard Saum	BRL (NEL)	Edgewood Arsenal, Md.
21.	William Ralph	NWL	Dahlgren, Virginia
22.	Robert E. Gray	NWL	Dahlgren, Virginia
23.	Roy R. Hilbrand	ASD	Dayton, Ohio
24.	Gerald Bennett	ASD	Wright-Patterson AFB, Ohio
25.	Robert Kesselman	Picatinny Arsenal	Dover, New Jersey
26.	Joe Burgio	Picatinny Arsenal	Dover, New Jersey
27.	Norman S. Banks	BRL (TBL)	APG, Md.
28.	Robert Lake	AMSAA (ASA)	APG, Md.
29.	Ronald Marking	AMSAA (ASA)	APG, Md.

Next page is blank.

APPENDIX B MAGI REPORT MR 6902 (ABRIDGED)

CONTENTS

		Page
B.1	ORGANIZATION OF THE NEW MAGI PROGRAM	47
B.1.1	Changes to CALC	47
B.1.2	Changes to GENI	49
B.1.3	Changes to G1	49
B.1.4	Changes to WOWI	49
B.1.5	Addition of TEC	49
B.1.6	Addition of TOR	49
B.1.7	Addition of QRTIC	49
B.1.8	Addition of CUBIC	50
B.1.9	Addition of DOT	50
B.1.10	Addition of CROSS	50
B.1.11	Addition of UNIT	50
B.1.12	Addition of ARS	50
B.2	DESCRIPTION OF INPUT PARAMETERS	52
B.2.1	Truncated Elliptic Cone	52
B.2.2	Torus	52
B.2.3	Arbitrary Surface	52
B.3	BODY CARDS	55
B.4	NEW VARIABLES IN COMMON	55
B.5	NEW ERROR STOPS	56

Next page is blank.

B.1 ORGANIZATION OF THE NEW MAGIC PROGRAM

The general organization of the MAGIC program remains as before. The following subroutines were changed: CALC, GENI, G1, WOWI. The following routines were added: TEC, TOR, ARS, DOT, CROSS, UNIT, QRTIC, CUBIC, ARIN. A description of the changes to existing subroutines and the new subroutines follows:

B.1.1 Changes to CALC.

CALC - Statement 18 - Change the computed "go to" by adding locations for computing the normals in the new bodies.

Statement 500

This section computes the normals to the elliptic cone. If the point of intersection was the top or bottom the normal is the same as the normal received as input. If the point is on the side of the cone the following equation is used.

$$(1) \qquad \overline{W}_B = \frac{(\ )_1}{\tau}\,\overline{A}^* - \frac{\overline{H}\cdot\overline{A}^*}{\overline{H}\cdot\overline{\eta}}\,\overline{\eta} + \frac{(\ )_2}{1}\,K^* - \frac{\overline{H}\cdot\overline{K}^*}{\overline{H}\ \overline{\eta}}\,\overline{\eta} - m(r_4-r_2)\,\frac{\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\,,$$

where $$(\ )_1 = \left(\overline{X} - \frac{\overline{X}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\,\overline{H} - \overline{V} + \frac{\overline{V}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\,\overline{H}\right)\cdot\overline{A}^*, \text{ and}$$

$$(\ )_2 = (\ )_1 \text{ with } \overline{K}^* \text{ instead of } \overline{A}^*$$

The terms are defined as follows:

$\overline{H}$ = height vector of cone,

$\overline{A}^*$ = unit vector - direction of major axis,

$\overline{K}^*$ = unit vector - direction of minor axis,

τ = ratio of major to minor axis squared,

$$m = \frac{(\overline{X}-\overline{V})}{\overline{H}\cdot\overline{\eta}}\cdot\overline{\eta}\,(r_4-r_2) + r_2$$

$\overline{X}$ = point of intersection,

$\overline{\eta}$ = normal to cutting plane, and

$\overline{V}$ = center of base ellipse.

The derivation is shown on the following page.

The equation for the ellipse parallel to the base ellipse and through the point of intersection $\overline{X}$ is

(2) $$\frac{((\overline{X}-\overline{V}-\gamma\overline{H})\cdot\overline{A}^*)^2}{\tau m^2} + \frac{((\overline{X}-\overline{V}-\gamma\overline{H})\cdot\overline{K}^*)^2}{m^2} = 1 ,$$

(3) $$\gamma = \frac{(\overline{X}-\overline{V})\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}} , \text{ and}$$

(4) $$m = \gamma r_4 + (1-\gamma) r_2 .$$

NOTE: γ and τ and $\overline{X}$ are known.

Expanding we get

$$0 = \left(\frac{\overline{X} - \frac{\overline{X}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\overline{H} - \overline{V}\frac{\overline{V}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\overline{H}}{\tau}\cdot\overline{A}^*\right)^2 + \left(\frac{\overline{X} - \frac{\overline{X}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\overline{H} - \overline{V}\frac{\overline{V}\cdot\overline{\eta}}{\overline{H}\cdot\overline{\eta}}\overline{H}}{1}\cdot\overline{K}^*\right)^2 - m^2.$$

Differentiating, with respect to X, Y, and Z, and taking the unit vector of the result gives us Equation (1).

Statement 550

The section computes the normal to the torus at the point of intersection. The equation used is:

(1) $$\overline{W}_B = \frac{\overline{X} - (\overline{C}+r_1\overline{d}^*)}{r_2}$$

where $\overline{X}$ is point of intersection,

$\overline{C}$ is center of torus,

r_1 is distance from center to the locus of mid-point of circular cross section

$\overline{d}^*$ = unit $\{\overline{d}\}$,

$\overline{d} = \{\overline{\eta}X(\overline{X}-\overline{c})\} X \overline{\eta}$ = direction of r_1 in plane, and

r_2 is radius of circular cross section.

Statement 575

This coding examines the intersection with the arbitrary surface (XI), selects the proper normal from those left in the MASTER/ASTER array by ARS, and places this normal into WB. See write-up of ARS routine to determine details of computation.

B.1.2 Changes to GENI.

GENI - Change all computed GO TO's involving body type to add three new bodies. Add conversion routines to store TORUS and Elliptic Cone data in either floating point or triplet and scalar form. Add coding to store Arbitrary Surface in floating point format. (Subroutine ARIN).

B.1.3 Changes to G1.

G1 - For a new ray, processing remains the same, except for changing the computed GO TO to check the new bodies. For continuation of a ray, a check is made for TORUS or ARBITRARY SURFACE. If the body is neither of these, the previously computed value is used. However, if the body type is one of these a check is made to see if the distance we have traveled is greater than ROUT. If it is not then we use the existing values for RIN and ROUT. Otherwise, we reenter the body routine and compute the next RIN/ROUT set (if any).

B.1.4 Changes to WOWI.

WOWI - The same changes as were made to G1 apply here. A description of the new routines follow.

B.1.5 Addition of TEC.

TEC - Body routine for truncated elliptic cone. Computes RIN, ROUT, LRI, LRO for intersection of ray and cone. Uses DOT, CROSS, and SQRT.

B.1.6 Addition of TOR.

TOR - Body routine for torus. Computes RIN and ROUT; LRI and LRO are 1. If four roots are found it selects the first pair greater than DIST as RIN and ROUT. Uses QRTIC and CUBIC.

B.1.7 Addition of QRTIC.

QRTIC (A,B,C) - Solves quartic polynomial equation with unit leading coefficient, $X^4 + C_1X^3 + C_2X^2 + C_3X + C_4 = 0$.

Method is from J. V. Uspensky, "Theory of Equations," pp 94-95. Used by TOR.

A = Array of coefficients

B = Array of roots

C = Number of real roots

B.1.8 Addition of CUBIC.

CUBIC (A,B,C) - Solves cubic polynomial equation with unit leading coefficient, $X^3 + C_1X^2 + C_2X + C_3 = 0$.

Method is from J. V. Uspensky, "Theory of Equations," pp 84-93. Used by TOR.

A = Array of coefficients

B = Array of roots

C = Number of real roots

B.1.9 Addition of DOT.

DOT (A,B) - Computes the dot product of vectors $\overline{A}$ and $\overline{B}$.

B.1.10 Addition of CROSS.

CROSS (A,B,C) - Computes the cross product of vectors $\overline{B}$ and $\overline{C}$ and stores result in vector $\overline{A}$.

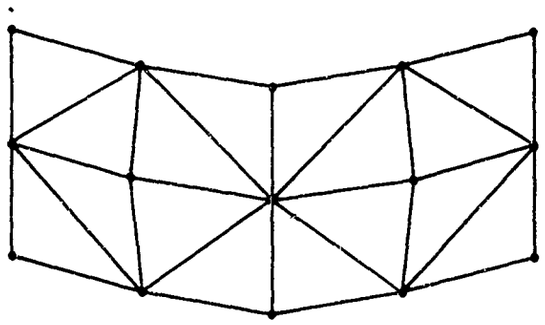
B.1.11 Addition of UNIT.

UNIT (A) - Computes unit vector of $\overline{A}$ and stores back in $\overline{A}$.

B.1.12 Addition of ARS.

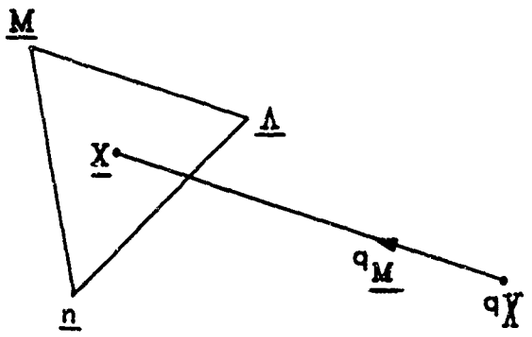
ARS - Body routine for arbitrary surface. Computes RIN, ROUT. LRI and LRO are always 1.

For the purposes of determining intersections and normals, the routine constructs a series of triangles, as below, and utilizes these triangles to determine intersections and the associated normals.



When entered for a new ray, the intersections are stored in the MASTER/ASTER array, together with the normals at these points. Upon reentry for the same ray, the RIN, ROUT pair appropriate to DIST are selected and returned to the calling routine.

To determine the intersection and normals, the ray is tested against each triangle for which at least one point of the triangle lies within the projection of the grid square. The calculation is done as follows:



By the rules of convex figures in 2-space, for $\bar{x}$ within the triangle, there must exist α,β,γ such that

(1) $$\alpha\bar{u} + \beta\bar{v} + \gamma\bar{w} = \bar{x} = \bar{x}_b + s\bar{w}_b$$

$$\alpha + \beta + \gamma = 1;\ \alpha,\beta,\gamma > 0\ ;$$

then

$$\gamma = 1 - \alpha - \beta\ ,$$

(2) $$\alpha\bar{u} + \beta\bar{v} + (1-\alpha-\beta)\bar{w} = \bar{x}_b + s\bar{w}_b\ ,\ \text{and}$$

(3) $$\alpha(\bar{u}-\bar{w}) + \beta(\bar{v}-\bar{w}) - s\bar{w}_b = \bar{x}_b - \bar{w}\ .$$

These are simply three equations in three unknowns.

Using determinants to solve this set of simultaneous equations we obtain α,β,γ, and s. After verifying that $\alpha + \beta + \gamma = 1$ and $\alpha,\beta,\gamma \geq 0$, we record the s value as well as a unit normal to the triangle. If α,β,γ fail to meet these requirements, the ray missed this triangle.

After performing the calculations for each triangle, the resulting s values and normals are placed in the MASTER/ASTER array. Sufficient space is provided for up to ten pairs of RIN and ROUT. The variable DIST is used to determine which pair of RIN and ROUT should be returned to the calling routine.

B.2 DESCRIPTION OF INPUT PARAMETERS

B.2.1 Truncated Elliptic Cone.

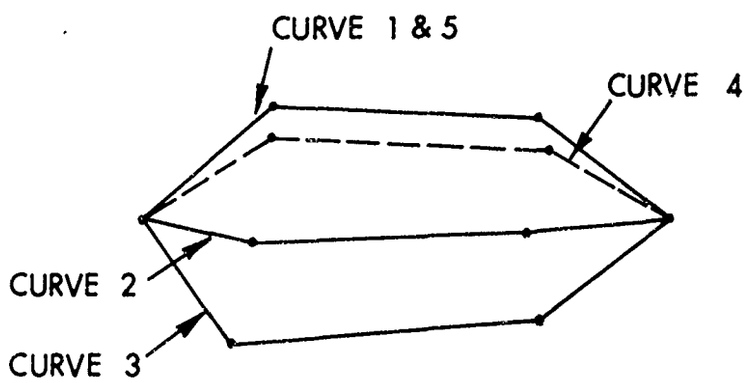
Specify a vertex V at the center of the larger ellipse, the height vector H, expressed in terms of its x,y,z components, the direction of the major axis A, the direction of the normal N, and three scalars - R1, the length of the major axis of the larger ellipse, R2, the length of the minor axis of the larger ellipse, and P, the ratio of the larger to the smaller ellipse.

B.2.2 Torus.

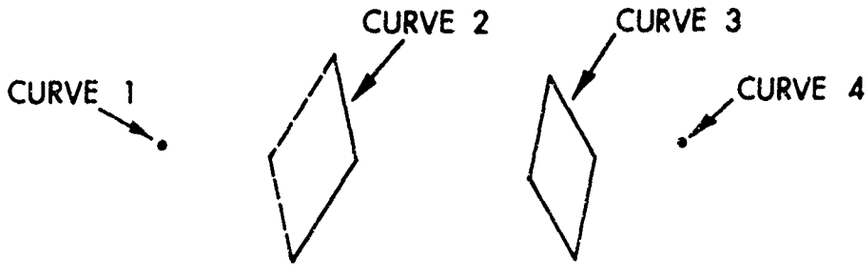
Specify a vertex V at the center of the torus, the normal to the plane in which the locus of mid-points of the circular cross sections lie, N, and two scalars - R1, the distance from the center to the mid-point of circular cross section, and R2, the radius of circular cross section.

B.2.3 Arbitrary Surface.

Specify the number of curves (M) to be specified and the number of points (N) which will be specified on each curve. A surface is constructed between curve 1 and curve 2, between curve 2 and curve 3, etc. The user must be careful that the described figure is closed and solid.



In the previous example, the first and last point of each curve is identical, and the first curve is identical to the last, and one can see that the figure is solid.



In this example, curve 1 consists of the same point repeated five times, curve 2 of five points (the first and fifth point being the same), curve 3 is defined similar to curve 2, and curve 4 similar to curve 1. It can also be seen that this figure is solid.

To further illustrate this figure, note in the figure that

M = 4	N = 5
Curve 1	pt. A_1
	B_1
	C_1
	D_1
	E_1
Curve 2	pt. A_2
	B_2
	C_2
	D_2
	E_2

M = 4	N = 5	
Curve 3	pt.	A_3
		B_3
		C_3
		D_3
		E_3
Curve 4	pt.	A_4
		B_4
		C_4
		D_4
		E_4

is identically the same figure as

M = 5	N = 4	
Curve 1	pt.	A_1
		A_2
		A_3
		A_4
Curve 2	pt.	B_1
		B_2
		B_3
		B_4
Curve 3	pt.	C_1
		C_2
		C_3
		C_4

M = 5	N = 4
Curve 4	pt. D_1
	D_2
	D_3
	D_4
Curve 5	pt. E_1
	E_2
	E_3
	E_4

The user should use this isomorphism as a check on whether the figure defined is truly closed and solid.

B.3 BODY CARDS

The data describing each body must be input using the format described in Table B.1. This table is similar to Table 3.1 (page 37 in the original document) and should be viewed as an extension of that table.

B.4 NEW VARIABLES IN COMMON

Variable Name	Labeled Common	Definition
IGRID	DAVIS	The grid square of the origin of the current ray (XBS)
LOOP	DAVIS	Set by G1 upon entry to a body routine to reflect ray number of last ray fired at the body.
INORM	DAVIS	Set by G1 to indicate if the ray is being fired normal to a surface (normal is computed by CALC)

B.5 NEW ERROR STOPS

Routine	Message	Explanation
CALC	ARS DID NOT FIND NORMAL	Data in MASTER/ASTER Array inconsistent. Some routine has destroyed portions of MASTER/ASTER.

TABLE B.1

Body Type	3 Letter ID	11-20	21-30	31-40	41-50	51-60	61-70	No. of Cards Needed
Truncated Elliptic Cone	TEC	V_x	V_y	V_z	H_x	H_y	H_z	1 of 3
		N_X^*	N_Y^*	N_Z^*	A_X^*	A_Y^*	A_Z^*	2 of 3
		R1	R2	P				3 of 3
Torus	TOR	V_x	V_y	V_z	N_x	N_y	N_z	1 of 2
		R1	R2					2 of 2
Arbitrary Curved Surface	ARS							
		M	N					1 of n
		X(1,1)	Y(1,1)	Z(1,1)	X(1,2)	Y(1,2)	Z(1,2)	3 of n
		⋮						
		X(1,N)	Y(1,N)	Z(1,N)				N+2 of n
		X(2,1)						
		⋮						
		X(M,1)						
		⋮						
		X(M,N)	Y(M,N)	Z(M,N)				

Next page is blank.

APPENDIX C

SAM-C BENCHMARK PROBLEM INPUT

Next page is blank.

C.1 SHIELDING BENCHMARK PROBLEM 4.0 (Abridged)

Gamma-Ray Dose Above a
Plane Source of ^{60}Co on an
Air/Ground Interface

Original Submitted by:

Charles W. Garrett
Radiation Research Associates, Inc.*
8404 Westmont Court
Bethesda, Maryland 20034

2 December 1968

Accepted by BPG:

December, 1968

This Abridged Version prepared September, 1969

*Corporate offices located in Fort Worth, Texas.

Next page is blank.

"1. INTRODUCTION

The dose rate three feet above an air/ground interface contaminated with gamma-ray emitting isotopes is often used as a basic normalizing parameter in fallout radiation environment studies and fallout shielding methodology. For example, the dose rate three feet above a fallout field is the basic quantity to which geometric and barrier factors are applied in the currently-used "Engineering Method" (References 1,2*). This technique predicts shield effectiveness in fallout situations. There have been experimental measurements of the dose above fallout fields in several weapons test (References 3,4) and several calculations (References 1,5) of the same quantity have been made.

"However, because of the obvious difficulties associated with measurements of actual fallout, many studies have concerned themselves with the radiation environment above interface planes contaminated with a single isotope. In particular, ^{137}Cs and ^{60}Co have been extensively used in these investigations.

"2. THE AIR/GROUND INTERFACE PROBLEM

This benchmark problem is concerned with the theoretical computation of various quantities above an ideal air/ground interface uniformly contaminated with ^{60}Co. A discussion of experimental results is also included for comparative purposes.

"2.1 Problem Geometry

Figure C1 illustrates the geometry for the theoretical benchmark. A receiver (detector) point is located three feet above the air/ground interface which is assumed to be smooth and infinite in extent. ^{60}Co is uniformly distributed on the interface. Although ^{60}Co emits one 1.17 MeV photon and one 1.33 MeV photon per disintegration, many studies assume an average photon energy of 1.25 MeV. This assumption introduces negligible errors, and the benchmark data are normalized to a source strength of one 1.25 MeV photon emitted isotropically per cm^2 of interface area per second. [For SAM C the photon energy of 1.33 MeV was used.]

---

*References are not included in this abridged version. Ed.

BASED ON ORNL-DWG 69-2121 IN SHIELDING BENCHMARK PROBLEM 4.0.

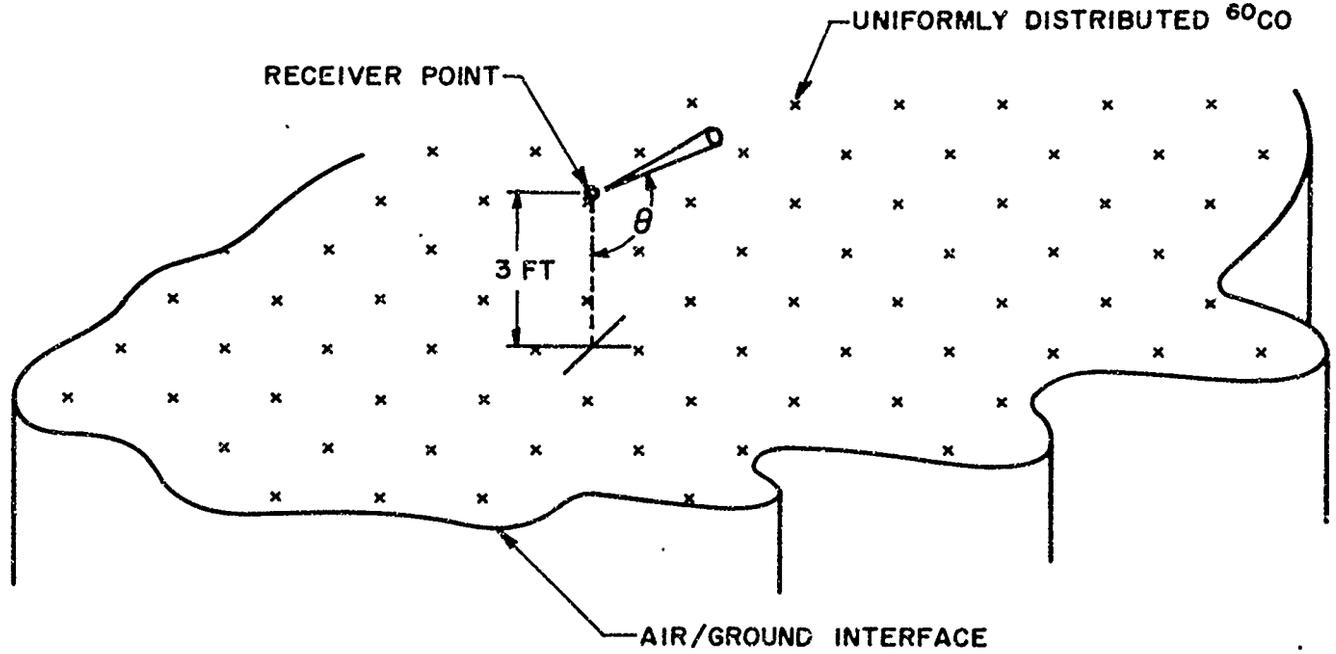


Figure CI. Air Over Ground Geometry

"Table I lists the constituents of the air and ground. The air density is 1.29×10^{-3} g/cm^3; the ground is assumed to have a composition similar to Nevada Test Site soil (Reference 6).

TABLE I AIR AND GROUND COMPOSITIONS

	Element	Atomic Concentration (atoms/cm^3)
Air*	Nitrogen	4.19×10^{19}
	Oxygen	1.13×10^{19}
	Argon	2.53×10^{17}
Ground	Hydrogen	8.55×10^{21}
	Oxygen	2.27×10^{22}
	Aluminum	2.01×10^{21}
	Silicon	9.53×10^{21}

*There has been a slight change in format. Ed.

"2.2 Quantities Calculated

Quantities calculated at the receiver point are: (1) the total kerma rate,** T, in air; (2) the kerma rate,** D, in air from uncollided photons; (3) the dose buildup factor, B; and (4) the differential angle and energy distribution of the number flux density, $\Phi(E,\theta)$, where θ is the receiver polar angle (Figure C1). The number flux density energy spectrum, I(E), [is the only quantity calculated by SAM C] ...."[calculated results appear in Tabular Form in Table IV and in graphical form in Figure C2].

**For the photon energies and geometry of this problem, the numerical difference between kerma rate and absorbed dose rate in air is small and can be ignored. Other studies quote kerma rate in tissue, and adjustments should be made to compare the results of such studies with this benchmark.

TABLE IV* SCATTERED PHOTON KERMA RATES THREE FEET ABOVE AN INFINITE ^{60}Co CONTAMINATED PLANE

Energy Interval (MeV)	K(E) (ergs=cm^2/g)	Flux Density (Photons/cm^2-sec)	Kerma Rate (ergs/g-sec)
.02 - .03	1.06(-8)	1.47(-3)	1.56(-11)
.03 - .04	5.28(-9)	1.55(-2)	8.18(-11)
.04 - .06	3.06(-9)	9.32(-2)	2.85(-10)
.06 - .10	3.06(-9)	2.10(-1)	6.43(-10)
.10 - .18	5.56(-9)	2.99(-1)	1.66(- 9)
.18 - .30	1.08(-8)	2.94(-1)	3.18(- 9)
.30 - .50	1.89(-8)	1.93(-1)	3.65(- 9)
.50 - .75	2.92(-8)	1.08(-1)	3.15(- 9)
.75 - 1.00	3.97(-8)	8.31(-2)	3.30(- 9)
1.00 - 1.25	5.00(-8)	1.69(-1)	8.45(- 9)
	TOTALS:	1.47(0)	2.44(-8)

NOTE: Read 1.06(-8) as 1.06×10^{-8}

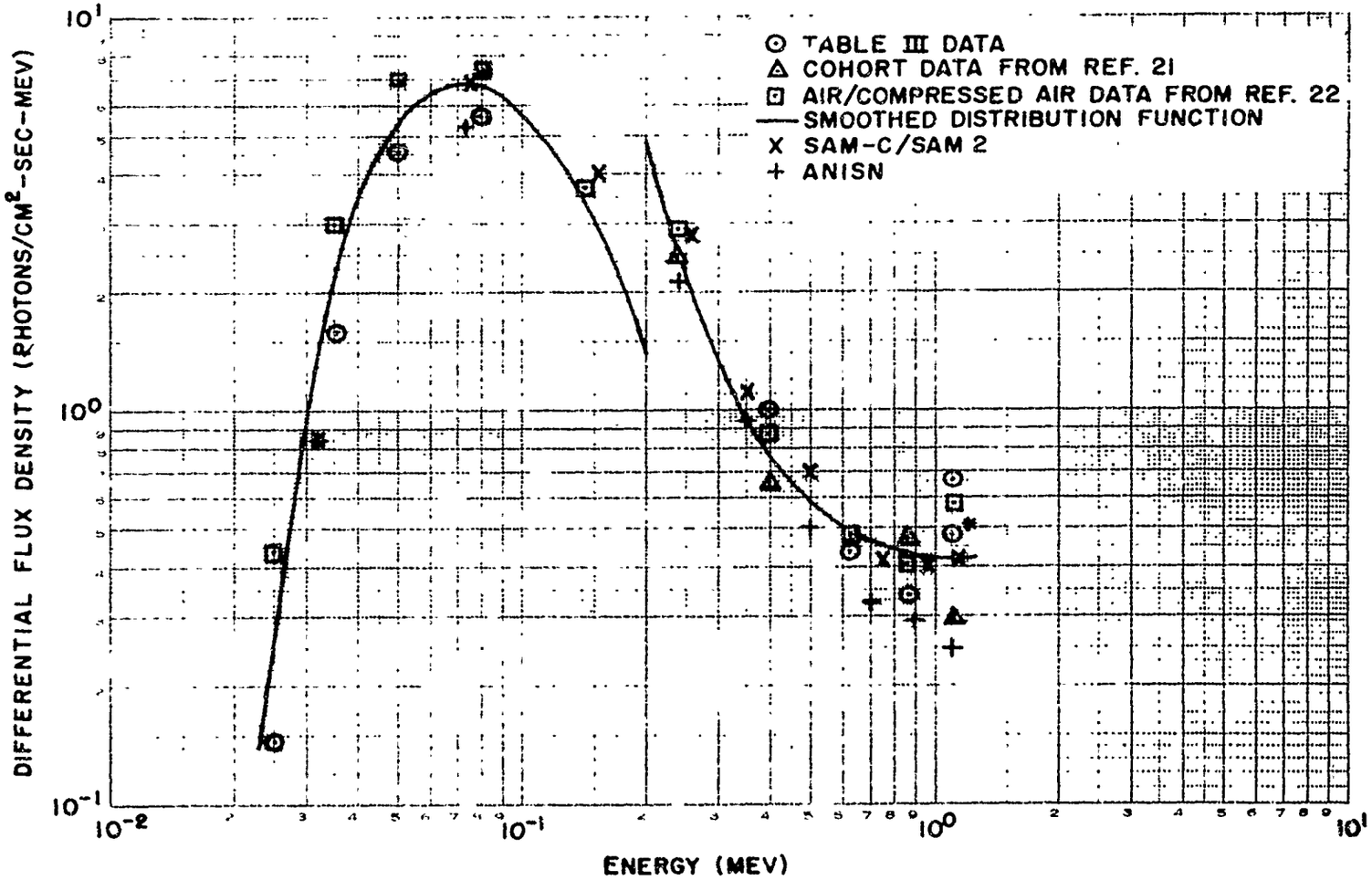
*Tables II and III have been omitted in this version. Ed.

[Results]

"On Figure C2 (Figure 6 in original document), differential scattered photon flux density energy spectra are plotted for the two cases shown on Figure 4,* along with data from a 7,000 history air/ground COHORT Monte Carlo study by French (Reference 21). Although neither the Table III or the COHORT data was smoothed in any way prior to constructing Figure 5, adjustments had to be made in the three lowest energy groups of the air/compressed air case. In that case, a severe fluctuation (visible on Figure 4) occurred in the 30° - 40° angle interval in each of the three energy bins. The solid curve on Figure 5 was obtained by intuitively smoothing all available data and, in addition, making use of two known facts; the magnitude of the discontinuity at the first scattering cutoff energy (0.212 MeV) and the value at the source energy (1.25 MeV). As described in Reference 24, these values can be easily and accurately computed. The value computed for the energy spectrum at 1.25 MeV is 0.43 photons/cm^2-sec-MeV, and the magnitude of the discontinuity is 3.40 photons/cm^2-sec-MeV.

*Figures 2, 3, 4, and 5 of original document have been omitted. Ed.

BASED ON ORNL-DWG 69-3429 IN SHIELDING BENCHMARK PROBLEM 4.0



* Plotted Incorrectly on Original.

Figure C2 Monte Carlo Scattered Flux Density Energy Spectra.

"In conclusion, it is to be emphasized that the differential data and smoothed curves presented on Figures 3, 4, and 5 contain rather large uncertainties, and must not be taken as absolute standards...."

C.2 AIR FORCE WEAPON LAB "ROCKET" GEOMETRY

Next page is blank.

FORTRAN CODING FORM AND DATA SHEET

PROGRAM SAMC GEOMETRY INPUT

PROGRAMMER A KRIS WIDDISON

```
TEST GEØMETRY -- INCLUDES ALL TYPES ØF SØLIDS
                I          I          I          II         II
 -250 0        150 0      -50 0      450 0
 -50 0         250 0      450 0
   2.0
*  RAW          0 0        0 0        0.0        0 0        0 0       20 0
                0 0       60.0        0.0      100 0        0 0        0 0
*  BØX          0.0        0 0        0 0        0 0        0 0       20 0
                0 0       60 0        0 0      -80 0        0 0        0 0
*  REC       -100 0       30 0        0.0        0.0        0 0       20 0
                0 0       15 0        0 0       10 0        0 0        0 0
*  RCC        -30.0       40.0       20.0        0 0        0 0      150 0
              10.0
*  TRC        -30.0       40 0      170.0        0.0        0 0       10 0
              10 0        0 0
*  ARB        -20 0        0 0       20.0      -20 0       20 0       20 0
              -20 0        0 0      220 0      -20 0       20.0      220 0
              -60 0        0 0       20.0      -60 0       20 0       20 0
              -60 0        0 0      220.0      -60.0       20 0      220 0
         1234 2684 6578 1375 3487 1562
*  SPH        -30.0       40.0      172.0        1.0
*  ELL        -30.0       40.0      140.0      -30.0       30.0      150 0
              20.0
*  SPH        -30.0       40.0       30.0        5.0
```

FORTRAN CODING FORM AND DATA SHEET

PROGRAM	SAM-C GEOMETRY (CONT.)	LAB	0 = ZERO	1 = ONE	2 = TWO	PAGE
PROGRAMMER	A. KRIS WIDDISON	PHONE / DATE	Ø = ALPHA O	I = ALPHA I	Z = ALPHA Z	

STATEMENT NUMBER	CONT	FORTRAN STATEMENT	IDENTIFICATION
SPH		1	
* ARB		-19.0 29.0 140.0 -41.0 29.0 140.0	
		-41.0 29.0 165.0 -19.0 29.0 165.0	
		-48.0 252.0 445.0 -52.0 252.0 445.0	
		-52.0 252.0 455.0 -48.0 252.0 455.0	
		1234 4378 5678 1265 1485 2376	
11		+ 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9	
		- 10 - 11 - 12	
2		+ 2 - 3	
3		+ 3 - 2 - 5 - 7	
4		+ 4	
5		+ 5 - 3 - 6 - 9 - 10	
6		+ 6 - 5 - 8	
7		+ 7 - 3	
8		+ 8	
9		ØR 9ØR 10	
10		+ 11	
11		+ 12 - 5 - 6 - 7 - 8 - 9 - 11	
-1			

AMORD-C Form 15 20 May 70 (Replaces AMORD-C Form 13 9 Mar 66)

APPENDIX D

LISTING OF THE AMSAA OCTOBER REVISED STANDARD MAGIC PROGRAM

Next page is blank.

```
C
      DIMENSION MASTER(30000),A(6)
      COMMON ASTER(30000)
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
      COMMON/TEMPOR/XS(6),X(6),IX(8),IT(10),IA(9),IN(9)
      COMMON/WALT/LIREC,NGIERR
      COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPEB,NO,IYES
      COMMON/ENGEOM/LEGEOM
      COMMON/SIZE/NDQ
      COMMON/ERR/IERR
      COMMON/RANDM/IRANDM
      EQUIVALENCE (ASTER,MASTER)
C
  901 FORMAT(1H1,32HTHIS IS THE 11 APR 69 VERSION OF /
     1        1H ,32HTHE BRLESC MAGIC PROGRAM  ****** //)
  902 FORMAT(16H BEGIN EXECUTION)
  903 FORMAT(8I10)
  904 FORMAT(1H0,10X,42HTHE TAPE 4 USED FOR THIS RUN HAS THE TITLE /
     1  10A6/)
  905 FORMAT(1H0,10HENTER GENI)
  906 FORMAT(1H0,12HLEAVING GENI)
  907 FORMAT(1H0,35HTERMINATION ON GEOMETRY INPUT ERROR,5X,5HIERR=,I5)
  906 FORMAT(1H1,15HTESTG IS CALLED)
  909 FORMAT(1H0,13HLEAVING TESTG)
  910 FORMAT(1H1,24HREGION TYPE DATA FOLLOWS, 8X,6HLIREO=,I10/
     1        1H ,6HREGION,6X,4HCODE,6X,4HTYPE,6X,11HDESCRIPTION/)
  911 FORMAT(3I10,10X,6A6)
  912 FORMAT(I6,I10,I9,7X,6A6)
  913 FORMAT(1H0,23HNO ROOM FOR IDENT TABLE,5X,7HLEGEOM=,I7,5X,
     1    6HLIREO=,I7)
  914 FORMAT(1H0,32HWRITE TAPE 1 OPTION IS SPECIFIED)
  915 FORMAT(I5,10X,10A6)
  916 FORMAT(1H1,11HENTER VOLUM)
  917 FORMAT(1H0,13HLEAVING VOLUM)
  918 FORMAT(1H ,6H 999.9)
  919 FORMAT(1H1,11HEND OF CASE,I5)
  925 FORMAT(1H1,32HNUM OF ASPECT ANGLES FOR GRID IS,I5)
  927 FORMAT(10I5)
  928 FORMAT(1H1,32HNUM OF ASPECT ANGLES FOR AREA IS,I5)
  929 FORMAT(1H0,31HNUMBER OF G1 ERRORS ENCOUNTERED,I5)
  930 FORMAT(1H0,31HNUMBER OF  0  ITEMS ENCOUNTERED,I5)
  999 FORMAT(1H0,10HEND OF RUN)
C
      IRANDM=0
      WRITE (6,901)
      WRITE (6,902)
C
      I15=2**15
      I30=2**30
      PINF=1.0E50
      NO=0
      IYES=1
      IERR=0
      LBASE=1
      KLOOP=0
      NDQ=30000
C
      READ (5,903)IROTP4,IWRIP4,ITESTG,IRAYSK,ICARDI,IENTLV,IVOLUM
```

```
      IF(IRDTP4.NE.0)IRDTP4=IYES
      IF(IWRTP4.NE.0)IWRTP4=IYES
      IF(ITESTG.NE.0)ITESTG=IYES
      IF(IRAYSK.NE.0)IRAYSK=IYES
      IF(ICARDI.NE.0)ICARDI=IYES
      IF(IENTLV.NE.0)IENTLV=IYES
      IF(IVOLUM.NE.0)IVOLUM=IYES
C
      IF(IRDTP4.EQ.NO) GOTO 10
      READ (4) DUMMY,ASTER,LBASE,PINF,IERR,NRPP,NTRIP,NSCAL,
     1  NBODY,NRMAX,LTRIP,LSCAL,LREGD,LDATA,LRIN,LROT,LIO,LOCDA,
     2  LBODY,LIRFO,SCALE,LRI,LRO,PINF,IT
      WRITE (6,904)(IT(I),I=1,10)
      GOTO 20
C
C      PROCESS GEOMETRY
C
   10 DO 11 I=LBASE,NDQ
      ASTER(I)=0.
   11 CONTINUE
C
      WRITE (6,905)
      CALL GENI
      WRITE (6,906)
      IERR=0
      IF(IERR.LE.0)GOTO 12
      WRITE (6,907)IERR
      STOP
C
   12 IF(IWRTP4.EQ.NO)GOTO 20
      WRITE(4) DUMMY,ASTER,LBASE,PINF,IERR,NRPP,NTRIP,NSCAL,
     1  NBODY,NRMAX,LTRIP,LSCAL,LREGD,LDATA,LRIN,LROT,LIO,LOCDA,
     2  LBODY,LIRFO,SCALE,LRI,LRO,PINF,IT
C
C     TEST G
C
   20 IF(ITESTG.EQ.NO)GOTO 30
      WRITE (6,908)
      CALL TESTG
      WRITE (6,909)
      ITESTG=NO
C
C     VOLUM
C
   30 IF(IVOLUM.EQ.NO)GOTO 40
      WRITE (6,916)
      CALL VOLUM
      WRITE (6,917)
      IVOLUM=NO
C
C     REGION TYPE DATA          / ICODE / IDENT /
C
C     IRN  = REGION NUMBER
C     ICODE = ITEM CODE
C     IDENT = SPACE CODE AND SPECIAL IDENTIFICATION
C         0 NO IDENT CODE
C         10,20,30,40,50,60,70,80,90 SPECIAL IDENTIFICATION
C             SKIRT=10    ARMOR=20    TARGET=30
C         -1,1-9,11-19,21-29,......,91-99 SPACE CODES
C
```

```
   40 LIRFO=NDQ-NRMAX-10                                                MAIN 121
      IF(LIRFO.GT.LEGEOM)GOTO 41                                        MAIN 122
      WRITE(6,913)LEGEOM,LIRFO                                          MAIN 123
      STOP                                                              MAIN 124
   41 WRITE (6,910)LIRFO                                                MAIN 125
      MASTER(LIRFO-1)=I15+1                                             MAIN 126
C                                                                       MAIN 127
   42 READ(5,911)IRN,ICODE,IDENT,(A(I),I=1,6)                           MAIN 128
      IF(IRN.LE.0)GOTO 50                                               MAIN 129
      WRITE (6,912)IRN,ICODE,IDENT,(A(I),I=1,6)                         MAIN 130
      IDENT=IDENT+1                                                     MAIN 131
      K=LIRFO+IRN-1                                                     MAIN 132
      MASTER(K)=ICODE*I15+IDENT                                         MAIN 133
      GOTO 42                                                           MAIN 134
C                                                                       MAIN 135
C     NOAA  = NUM OF ASPECT ANGLES FOR GRID                             MAIN 136
C     ITAPE8 IS THE SUPRESS PRINTER OPTION                              MAIN 137
C     IWOT IS WRITE OPTION FOR TAPE 1                                   MAIN 138
C     NAREA = NUM OF ASPECT ANGLES FOR AREA                             MAIN 139
C                                                                       MAIN 140
   50 READ (5,927)NOAA,IWOT,ITAPE8,NAREA                                MAIN 141
      IF(IWOT.NE.0)IWOT=IYES                                            MAIN 142
      IF(ITAPE8.EQ.0)GOTO 51                                            MAIN 143
      ITAPE8=NO                                                         MAIN 144
      GOTO 52                                                           MAIN 145
   51 ITAPE8=IYES                                                       MAIN 146
   52 IF(IWOT.EQ.NO)GOTO 60                                             MAIN 147
      REWIND 1                                                          MAIN 148
      WRITE (6,914)                                                     MAIN 149
      WRITE(1,915)NOAA,(IT(I),I=1,10)                                   MAIN 150
C                                                                       MAIN 151
C     GRID                                                              MAIN 152
C                                                                       MAIN 153
   60 IF(NOAA.LE.0)GOTO 70                                              MAIN 154
      WRITE (6,925)NOAA                                                 MAIN 155
C                                                                       MAIN 156
      DO 61 I=1,NOAA                                                    MAIN 157
      CALL GRID                                                         MAIN 158
      IF(IWOT.EQ.IYES)WRITE(1,918)                                      MAIN 159
      WRITE (6,919)I                                                    MAIN 160
      WRITE (6,929)IERR                                                 MAIN 161
      WRITE (6,930)IERRO                                                MAIN 162
      IERR=0                                                            MAIN 163
      IERRO=0                                                           MAIN 164
   61 CONTINUE                                                          MAIN 165
C                                                                       MAIN 166
C     AREA                                                              MAIN 167
C                                                                       MAIN 168
   70 IF(NAREA.LE.0)GOTO 99                                             MAIN 169
      WRITE (6,928)NAREA                                                MAIN 170
C                                                                       MAIN 171
      DO 71 I=1,NAREA                                                   MAIN 172
      CALL AREA                                                         MAIN 173
      WRITE (6,919)I                                                    MAIN 174
      IERR=0                                                            MAIN 175
   71 CONTINUE                                                          MAIN 176
C                                                                       MAIN 177
   99 WRITE (6,999)                                                     MAIN 178
      STOP                                                              MAIN 179
      END                                                               MAIN 180
```

```
C                                                                       MAIN 181
C                                                                       MAIN 182
      SUBROUTINE UN2(L,J1,J2)                                           ****   1
C         UNPACKS 2 ITEMS FROM THE MASTER-ASTER ARRAY                   UN2    2
      COMMON MASTER(30000)                                              UN2    3
      I3=MASTER(L)                                                      UN2    4
      J1=I3/32768                                                       UN2    5
      J2=I3-J1*32768                                                    UN2    6
      RETURN                                                            UN2    7
      END                                                               UN2    8
C                                                                       UN2    9
C                                                                       UN2   10
      SUBROUTINE UN3(L,J1,J2,J3)                                        ****   2
C        UNPACKS 3 ITEMS FROM G1 STORAGE                                UN3    2
      COMMON MASTER(30000)                                              UN3    3
      I3=MASTER(L)                                                      UN3    4
      I2=I3/32768                                                       UN3    5
      J1=I2/64                                                          UN3    6
      J2=I2-J1*64                                                       UN3    7
      J3=I3-I2*32768                                                    UN3    8
      RETURN                                                            UN3    9
      END                                                               UN3   10
C                                                                       UN3   11
C                                                                       UN3   12
      SUBROUTINE OPENK(L,J1,J2,J3)                                      ****   3
      COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,         OPENK  2
     1    ITR(200),CA,CE,SA,SE                                          OPENK  3
C                                                                       OPENK  4
C     UNPACKS 3 ITEMS FROM COMPONENT LINE OF SIGHT STORAGE ITR          OPENK  5
C     / SURFACE NUM / BODY NUM / NEXT REGION /                          OPENK  6
C                                                                       OPENK  7
      I3=ITR(L)                                                         OPENK  8
      I2=I3/4096                                                        OPENK  9
      J1=I2/4096                                                        OPENK 10
      J2=I2-J1*4096                                                     OPENK 11
      J3=I3-I2*4096                                                     OPENK 12
      RETURN                                                            OPENK 13
      END                                                               OPENK 14
C                                                                       OPENK 15
C                                                                       OPENK 16
      FUNCTION RAN(M)                                                   ****   4
      COMMON/RANDM/IRN                                                  RAN    2
C         GENERATES RANDOM NUMBERS                                      RAN    3
      RAN=URAN31(IRN)                                                   RAN    4
      RETURN                                                            RAN    5
      END                                                               RAN    6
C                                                                       RAN    7
C                                                                       RAN    8
      FUNCTION URAN31(I)                                                ****   5
      IF(I)20,10,20                                                     URAN31 2
   10 I=11111111                                                        URAN31 3
   20 J=I                                                               URAN31 4
      J=J*25                                                            URAN31 5
      J=J-(J/67108864)*67108864                                         URAN31 6
      J=J*25                                                            URAN31 7
      J=J-(J/67108864)*67108864                                         URAN31 8
      J=J*5                                                             URAN31 9
      J=J-(J/67108864)*67108864                                         URAN3110
      A1=J                                                              URAN3111
      I=J                                                               URAN3112
```

```
      URAN31=A1/67108864.                                               URAN3113
      RETURN                                                            URAN3114
      END                                                               URAN3115
C                                                                       URAN3116
C                                                                       URAN3117
      SUBROUTINE CROSS(ANSWER,FIRST,SECOND)                             ****   6
      DIMENSION ANSWER(3),FIRST(3),SECOND(3)                            CROSS  2
C         COMPUTES CROSS PRODUCT      ANSWER = FIRST X SECOND           CROSS  3
      ANSWER(1) = FIRST(2)*SECOND(3) - FIRST(3)*SECOND(2)               CROSS  4
      ANSWER(2) = FIRST(3)*SECOND(1) - FIRST(1)*SECOND(3)               CROSS  5
      ANSWER(3) = FIRST(1)*SECOND(2) - FIRST(2)*SECOND(1)               CROSS  6
      RETURN                                                            CROSS  7
      END                                                               CROSS  8
C                                                                       CROSS  9
C                                                                       CROSS 10
      FUNCTION DOT(FIRST,SECOND)                                        ****   7
      DIMENSION FIRST(3),SECOND(3)                                      DOT    2
C         COMPUTES DOT PRODUCT      DOT = FIRST . SECOND                DOT    3
      DOT = FIRST(1)*SECOND(1)+FIRST(2)*SECOND(2)+FIRST(3)*SECOND(3)    DOT    4
      RETURN                                                            DOT    5
      END                                                               DOT    6
C                                                                       DOT    7
C                                                                       DOT    8
      SUBROUTINE UNIT(V)                                                ****   8
      DIMENSION V(3)                                                    UNIT   2
C         COMPUTES UNIT VECTOR                                          UNIT   3
      TEMP = SQRT(DOT(V,V))                                             UNIT   4
      V(1)=V(1)/TEMP                                                    UNIT   5
      V(2)=V(2)/TEMP                                                    UNIT   6
      V(3)=V(3)/TEMP                                                    UNIT   7
      RETURN                                                            UNIT   8
      END                                                               UNIT   9
C                                                                       UNIT  10
C                                                                       UNIT  11
      SUBROUTINE QRTIC(C,R,NRE)                                         ****   9
C                                                                       QRTIC  2
C     SOLVES A POLYNOMIAL EQUATION OF THE TYPE                          QRTIC  3
C       X**4 + C(1)*X**3 + C(2)*X**2 + C(3)*X + C(4) = 0                QRTIC  4
C     THE COEFFICIENT OF X**4 IS ASSUMED TO BE 1                        QRTIC  5
C     R CONTAINS THE ROOTS                                              QRTIC  6
C     NRE CONTAINS THE NUMBER OF REAL ROOTS                             QRTIC  7
C     IF THERE ARE TWO REAL ROOTS THEY WILL BE IN R(1) AND R(2),        QRTIC  8
C       WITH THE COMPLEX ROOTS R(3) +- R(4)*I                           QRTIC  9
C     IF THERE ARE NO REAL ROOTS, THE COMPLEX ROOTS ARE                 QRTIC 10
C       R(1) +- R(2)*I AND R(3) +- R(4)*I                               QRTIC 11
C                                                                       QRTIC 12
      DIMENSION C(4),R(4),CP(3),Y(3)                                    QRTIC 13
      C1SQ=C(1)**2                                                      QRTIC 14
      CP(1)=-C(2)                                                       QRTIC 15
      CP(2)=C(1)*C(3)-4.*C(4)                                           QRTIC 16
      CP(3)=(4.*C(2)-C1SQ)*C(4)-C(3)**2                                 QRTIC 17
      CALL CUBIC(CP,Y,NRE)                                              QRTIC 18
      A=C1SQ/4.-C(2)+Y(1)                                               QRTIC 19
      B=.5*C(1)*Y(1)-C(3)                                               QRTIC 20
      D=.25*Y(1)**2-C(4)                                                QRTIC 21
      IF(A.GT.0.)GOTO 10                                                QRTIC 22
      E=0.                                                              QRTIC 23
      GOTO 20                                                           QRTIC 24
   10 E=SQRT(A)                                                         QRTIC 25
   20 IF(D.GT.0.)GOTO 30                                                QRTIC 26
```

```
      F=0.                                                              QRTIC 27
      GOTO 50                                                           QRTIC 28
   30 F=SIGN(SQRT(D),B)                                                 QRTIC 29
   50 NRE=0                                                             QRTIC 30
      REAL=-.25*C(1)+.5*L                                               QRTIC 31
      DSCR=REAL**2-.5*Y(1)+F                                            QRTIC 32
      RAD=SQRT(ABS(DSCR))                                               QRTIC 33
      IF(DSCR.LT.0.)GOTO 60                                             QRTIC 34
      NRE=2                                                             QRTIC 35
      R(1)=REAL+RAD                                                     QRTIC 36
      R(2)=REAL-RAD                                                     QRTIC 37
      GOTO 65                                                           QRTIC 38
   60 R(3)=REAL                                                         QRTIC 39
      R(4)=RAD                                                          QRTIC 40
   65 REAL=REAL-F                                                       QRTIC 41
      DSCR=REAL**2-.5*Y(1)-F                                            QRTIC 42
      RAD=SQRT(ABS(DSCR))                                               QRTIC 43
      IF(DSCR.LT.0.)GOTO 80                                             QRTIC 44
      NRE=NRE+2                                                         QRTIC 45
      R(NRE)=REAL-RAD                                                   QRTIC 46
      R(NRE-1)=REAL+RAD                                                 QRTIC 47
      RETURN                                                            QRTIC 48
   80 R(NRE+1)=REAL                                                     QRTIC 49
      R(NRE+2)=RAD                                                      QRTIC 50
      RETURN                                                            QRTIC 51
      END                                                               QRTIC 52
C                                                                       QRTIC 53
C                                                                       QRTIC 54
      SUBROUTINE CUBIC(C,R,NRE)                                         ****  10
C                                                                       CUBIC  2
C     SOLVES A POLYNOMIAL EQUATION OF THE TYPE                          CUBIC  3
C       X**3 + C(1)*X**2 + C(2)*X + C(3) = 0                            CUBIC  4
C     THE COEFFICIENT OF X**3 IS ASSUMED TO BE 1                        CUBIC  5
C     R CONTAINS THE ROOTS                                              CUBIC  6
C     NRE CONTAINS THE NUMBER OF REAL ROOTS                             CUBIC  7
C     IF THERE IS ONE REAL ROOT IT WILL BE IN R(1),                     CUBIC  8
C       WITH THE COMPLEX ROOTS R(2) +- R(3)*I                           CUBIC  9
C                                                                       CUBIC 10
      DIMENSION C(3),R(3)                                               CUBIC 11
      C1SQ=C(1)**2                                                      CUBIC 12
      P=C(2)-C1SQ/3.                                                    CUBIC 13
      Q=C(3)-C(1)*(C(2)/3.-2.*C1SQ/27.)                                 CUBIC 14
      DEL=4.*P**3+27.*Q**2.                                             CUBIC 15
      T=C(1)/3.                                                         CUBIC 16
      IF(DEL.LT.0.)GOTO 10                                              CUBIC 17
      SQ=SQRT(DEL/108.)                                                 CUBIC 18
      HQ=.5*Q                                                           CUBIC 19
      A=-HQ+SQ                                                          CUBIC 20
      B=-HQ-SQ                                                          CUBIC 21
      CRTA=SIGN(ABS(A)**.3333333333333333,A)                            CUBIC 22
      CRTB=SIGN(ABS(B)**.3333333333333333,B)                            CUBIC 23
      Y=CRTA+CRTB                                                       CUBIC 24
      R(1)=Y-T                                                          CUBIC 25
      R(2)=-.5*Y-T                                                      CUBIC 26
      R(3)=.866025404*(CRTA-CRTB)                                       CUBIC 27
      NRE=1                                                             CUBIC 28
      RETURN                                                            CUBIC 29
   10 PHI3=ATAN2(SQRT(-DEL/27.),-Q)/3.                                  CUBIC 30
      CON=2.*SQRT(-P/3.)                                                CUBIC 31
      R(1)=CON*COS(PHI3)-T                                              CUBIC 32
```

```
      R(2)=-CON*COS(1.04719755-PHI3)-T                                  CUBIC 33
      R(3)=-CON*COS(1.04719755+PHI3)-T                                  CUBIC 34
      NRE=3                                                             CUBIC 35
      RETURN                                                            CUBIC 36
      END                                                               CUBIC 37
C                                                                       CUBIC 38
C                                                                       CUBIC 39
      FUNCTION XDIST(XA,XB)                                             **** 11
C         COMPUTES THE DISTANCE BETWEEN XA AND XB                       XDIST  2
      DIMENSION XA(3),XB(3)                                             XDIST  3
      XSUM=0.                                                           XDIST  4
      DO 10 I=1,3                                                       XDIST  5
      XSUM=XSUM+(XA(I)-XB(I))**2                                        XDIST  6
   10 CONTINUE                                                          XDIST  7
      XDIST=SQRT(XSUM)                                                  XDIST  8
      RETURN                                                            XDIST  9
      END                                                               XDIST 10
C                                                                       XDIST 11
C                                                                       XDIST 12
      SUBROUTINE DCOSP(XA,XB,WA)                                        **** 12
C         COMPUTES DIRECTION COSINES FROM POINT XA TO POINT XB          DCOSP  2
C             AND STORES DIRECTION COSINES IN WA                        DCOSP  3
      DIMENSION XA(3),XB(3),WA(3)                                       DCOSP  4
      DIS=XDIST(XA,XB)                                                  DCOSP  5
      DO 10 I=1,3                                                       DCOSP  6
      WA(I)=(XB(I)-XA(I))/DIS                                           DCOSP  7
   10 CONTINUE                                                          DCOSP  8
      RETURN                                                            DCOSP  9
      END                                                               DCOSP 10
C                                                                       DCOSP 11
C                                                                       DCOSP 12
      SUBROUTINE TROPIC(WP)                                             **** 13
C         GENERATES RANDOM DIRECTION COSINES FROM AN                    TROPIC 2
C             ISOTROPIC DISTRIBUTION                                    TROPIC 3
      DIMENSION WP(3)                                                   TROPIC 4
   10 X1=RAN (-1)                                                       TROPIC 5
      X2=RAN (-1)                                                       TROPIC 6
      X1S=X1**2                                                         TROPIC 7
      X2S=X2**2                                                         TROPIC 8
      T=X1S+X2S                                                         TROPIC 9
      IF(T.GE.1.)GOTO 10                                                TROPIC10
C         CALC SIN AND COS OF A RANDOM ANGLE PHI                        TROPIC11
      CSPHI=(X1S-X2S)/T                                                 TROPIC12
      SNPHI=(2.*X1*X2)/T                                                TROPIC13
      X1=RAN (-1)                                                       TROPIC14
      IF(X1.LE..5)SNPHI=-SNPHI                                          TROPIC15
C         CALC COS AND SIN OF RANDOM ANGLE THT                          TROPIC16
      CSTHT=2.*RAN (-1)-1.                                              TROPIC17
      SNTHT=SQRT(1.-CSTHT**2)                                           TROPIC18
C         CALC DIRECTION COSINES                                        TROPIC19
      WP(1)=SNTHT*SNPHI                                                 TROPIC20
      WP(2)=SNTHT*CSPHI                                                 TROPIC21
      WP(3)=CSTHT                                                       TROPIC22
      RETURN                                                            TROPIC23
      END                                                               TROPIC24
C                                                                       TROPIC25
C                                                                       TROPIC26
C                                                                       TROPIC27
C                                                                       TROPIC28
      SUBROUTINE GENI                                                   **** 14
```

```
      DIMENSION MASTER(30000),ITY(11),IAN(8),IAA(8),FX(20),
     1  NOO(3),NO1(3),NO2(3),O4(3),TT(3),TT1(3),TT2(3),NBOD(11)
      COMMON ASTER(30000)
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
      COMMON/TEMPOR/XS(6),X(6),IX(8),IT(10),IA(9),IN(9)
      COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES
      COMMON/SIZE/NDQ
      COMMON/UNCLE/NN,IC(4)
      COMMON/RRPP/LRPPD,LABUT
      EQUIVALENCE(ASTER,MASTER)
C
  901 FORMAT(1H0,24HSTART READING SOLID DATA)
  902 FORMAT(10A6)
  903 FORMAT(1H0,10A6/)
  904 FORMAT(7I10)
  905 FORMAT(4X,34HNO. OF RECTANGULAR PARALLELEPIPEDS,I10/
     1       4X,34HNO. OF TRIPLETS                   ,I10/
     2       4X,34HNO. OF SCALERS                    ,I10/
     3       4X,34HNO. OF SOLIDS                     ,I10/
     4       4X,34HMAX NO. OF REGIONS                ,I10)
  906 FORMAT(1H0,45X,32HRECTANGULAR PARALLELEPIPED INPUT)
  907 FORMAT(1H0,37X,12HTRIPLET DATA)
  908 FORMAT(6E12.6)
  909 FORMAT(I8,17X,3F12.5)
  910 FORMAT(1H0,25X,12HSCALAR  DATA)
  911 FORMAT(1H0,50X,22HDESCRIPTION OF  SOLIDS)
  912 FORMAT(3A1,A3,A4,6F10.5)
  913 FORMAT(1H0,6HITYPE ,A3,27H DOES NOT MATCH WITH AN ITY)
  914 FORMAT(I9,1X,3A1,3X,A3,A4,3X,8I5)
  915 FORMAT(I8,1X,3A1,2X,A3,A4,4X,6F12.5)
  916 FORMAT(25X,6F12.5)
  917 FORMAT(1H0,38HNO MORE ROOM FOR SOLID DATA     LDATA=,I10,
     1  5X,5HLBOT=,I10,5X,4HNDQ=,I10)
  918 FORMAT(1H0,25HFINISH READING SOLID DATA)
  919 FORMAT(1H0,  5HLREGD,7H  LREGL,7H  LENLV,7H   LRIN,7H   LROT,
     1    7H    LIO,7H LEGEOM/I5,6I7)
  920 FORMAT(1H1,36X,23HREGION COMBINATION DATA)
  921 FORMAT(I5,1X,9(A2,I5))
  922 FORMAT(1H0,30HERROR IN DESCRIPTION OF REGION,I5,
     19H IN FIELD,I2,5X,24HBODY NUM.GT.NRPP + NBODY)
  923 FORMAT(10X,9(1H(,A2,I5,1H),1X))
  924 FORMAT(I8,2X,9(1H(,A2,I5,1H),1X))
  925 FORMAT(1H0,30HILLEGAL OPERATOR IN ABOVE CARD,5X,A2,
     1  9H IN FIELD,I2)
  926 FORMAT(1H0,29HERROR IN REGION INPUT    IR=,I5,14H OR N.GT.NRMAX)
  927 FORMAT(1H0,39HNO MORE ROOM FOR REGION DATA     LDATA=,I10,
     1  5X,4HNDQ=,I10)
  928 FORMAT(1H0,26HFINISH READING REGION DATA)
  929 FORMAT(14H ERROR, REGION,I10,18H IS PART OF REGION,I10)
  930 FORMAT(24H FINISH CHECKING REGION ,I5)
  931 FORMAT(1H0,34HNO MORE ROOM FOR ENTER LEAVE TABLE,5X,
     1  6HLDATA=,I10,5X,4HNDQ=I10,5X,4HPASS,I2,5X,3HIR=,I10)
  932 FORMAT(1H0,28HTOTAL ROOM FOR GEOMETRY DATA,5X,7HLEGEOM=,I6)
  933 FORMAT(1H0,5HENTER,18I6/(23X,15I6))
  934 FORMAT(1H ,5HLEAVE,18I6/(23X,15I6))
  935 FORMAT(1H1,50X,18HBEGIN ARRAY OUTPUT/)
  936 FORMAT(3(3I6,1X,E11.4,3H $ ))
  937 FORMAT(/)
```

```
  938 FORMAT(1H0,34HFINISH A PASS OF ENTER LEAVE TABLE,I5)
  939 FORMAT(1H0,14HERROR IN INPUT,5X,A3,23H DOES NOT ALLOW TRIPLET,
     1    22H AND SCALAR TYPE INPUT)
  940 FORMAT(10X,6F10.5)
  941 FORMAT(1H0,37HTERMINATION ON BAD REGION DESCRIPTION)
  942 FORMAT(1H0,32HERROR IN DESCRIPTION OF BODY NUM,I6/
     1    7H VECTOR,3F12.5,24H IS NOT PERPENDICULAR TO /
     2    7H VECTOR,3F12.5/)
  943 FORMAT(1H0,27HERROR IN DESCRIPTION OF TOR,5X,8HR2.GT.R1/)
  944 FORMAT(1H0,27HERROR IN DESCRIPTION OF TRC,5X,7HR1 = R2/)
  945 FORMAT(1H0,5HLBASE,7H  LRPPD,
     1    7H  LABUT,7H  LBODY,7H   LBOD,7H  LDATA,7H   LBOT,7H  LSCAL,
     2    7H  LTRIP,7H    NDQ/I5,9I7)
  946 FORMAT(1H1,17HENTER-LEAVE TABLE)
  947 FORMAT(1H0,11(2X,A3)/11I5)
  948 FORMAT(1H0,27HERROR IN DESCRIPTION OF TEC,5X,
     1    41HHEIGHT VECTOR IS PARALLEL TO BASE ELLIPSE)
C
      INTEGER HHBOX,HHSPH,HHRCC,HHREC,HHTRC,HHELL,HHRAW,HHARB,HHTEC,
     1 HHTOR,HHARS,HHOR,HHBR,HHR,HHRA,HHAR,HHBA,HHA,HHB
C
      DATA HHBOX,HHSPH,HHRCC,HHREC,HHTRC,HHELL,HHRAW,HHARB,
     1HHTEC,HHTOR,HHARS/3HBOX,3HSPH,3HRCC,3HREC,3HTRC,
     23HELL,3HRAW,3HARB,3HTEC,3HTOR,3HARS/
      DATA HHOR,HHBR,HHR,HHRA,HHAR,HHBA,HHA,HHB
     1/2HOR,2HBR,1HR,2HRA,2HAR,2H A,2HA ,2H  /
      ITY(1)=HHBOX
      ITY(2)=HHSPH
      ITY(3)=HHRCC
      ITY(4)=HHREC
      ITY(5)=HHTRC
      ITY(6)=HHELL
      ITY(7)=HHRAW
      ITY(8)=HHARB
      ITY(9)=HHTEC
      ITY(10)=HHTOR
      ITY(11)=HHARS
      IAN(1)=1
      IAN(2)=1
      IAN(3)=1
      IAN(4)=2
      IAN(5)=2
      IAN(6)=3
      IAN(7)=3
      IAN(8)=4
      IAA(1)=HHOR
      IAA(2)=HHBR
      IAA(3)=HHR
      IAA(4)=HHRA
      IAA(5)=HHAR
      IAA(6)=HHBA
      IAA(7)=HHA
      IAA(8)=HHB
      IBL=HHB
C
      WRITE (6,901)
      READ(5,902)(IT(I),I=1,10)
      WRITE  (6,903)(IT(I),I=1,10)
```

```
      READ(5,904)NRPP,NTRIP,NSCAL,NBODY,NRMAX,IPRIN,IRCHEK
      WRITE  (6,905)NRPP,NTRIP,NSCAL,NBODY,NRMAX
C
C     RPP
C
      WRITE (6,906)
      LAR=1
      IF(NRPP.LE.0)GOTO 20
      CALL RPPIN(LAR)
      IF(IERR.GT.0)RETURN
C
C     LBODY STORAGE      RESERVE 3*(NRPP+NBODY) WORDS
C         /      ITYPE       /      LDATA       /
C         / LOC ENTER LIST / LOC LEAVE LIST /
C         / NUM ENTER       / NUM LEAVE       /
C
C     LDATA POINTS TO BODY POINTERS STORED AT LBOD
C
   20 LTRIP=NDQ-3*NTRIP+1
      LSCAL=LTRIP-NSCAL
      LBOT=LSCAL
      L=LAR
      LBODY=L+1
      LDATA=LBODY+3*(NBODY+NRPP)
      LBOD=LDATA
C
C     TRIPLETS
C
      IF(NTRIP.EQ.0)GOTO 30
      WRITE (6,907)
      DO 21 I=1,NTRIP
      I1=LTRIP+3*(I-1)
      I2=I1+2
      READ(5,908)(ASTER(K),K=I1,I2)
      WRITE (6,909)(I,(ASTER(K),K=I1,I2))
   21 CONTINUE
C
C     SCALARS
C
   30 IF(NSCAL.EQ.0)GOTO 50
      I1=LSCAL
      I2=I1+NSCAL-1
      WRITE (6,910)
      DO 31 I=I1,I2
      J=I-I1+1
      READ(5,908)ASTER(I)
      WRITE (6,909)J,ASTER(I)
   31 CONTINUE
C
C     READ AND PROCESS BODIES
C
   50 WRITE (6,911)
C
C     LOOP TO PROCESS SOLIDS
C
      DO 370 N=1,NBODY
      NN=N+NRPP
      LS1=0
      READ(5,912) IC(1),IC(2),IC(3),ITYPE,IC(4),(FX(K),K=1,6)
      DO 51 I=1,11
```

```
      IF(ITYPE.EQ.ITY(I))GOTO 52                                        GENI 182
   51 CONTINUE                                                          GENI 183
      WRITE (6,913)ITYPE                                                GENI 184
      STOP                                                              GENI 185
   52 ITYPE=I                                                           GENI 186
      NBOD(I)=NBOD(I)+1                                                 GENI 187
      K=LBODY+3*(NRPP+N-1)                                              GENI 188
      MASTER(K)=ITYPE*I15+LDATA                                         GENI 189
      IF(IC(1).NE.IBL)GOTO 200                                          GENI 190
C                                                                       GENI 191
C          BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS                  GENI 192
      GOTO(101,103,102,101,101,102,101,100,104,101,100),ITYPE           GENI 193
  100 WRITE (6,939)ITY(ITYPE)                                           GENI 194
      STOP                                                              GENI 195
  101 LL=4                                                              GENI 196
      GOTO 110                                                          GENI 197
  102 LE=3                                                              GENI 198
      GOTO 110                                                          GENI 199
  103 LE=2                                                              GENI 200
      GOTO 110                                                          GENI 201
  104 LE=7                                                              GENI 202
  110 CALL CONVRT(FX,IX,LE)                                             GENI 203
      WRITE (6,914)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(IX(J),J=1,LE) GENI 204
      LT=LTRIP-3                                                        GENI 205
      J1=IX(1)                                                          GENI 206
      J2=IX(2)                                                          GENI 207
      J3=IX(3)                                                          GENI 208
      J4=IX(4)                                                          GENI 209
      J5=IX(5)                                                          GENI 210
      J6=IX(6)                                                          GENI 211
      J7=IX(7)                                                          GENI 212
C          BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS                  GENI 213
      GOTO(120,130,140,120,150,160,120,100,170,150,100),ITYPE           GENI 214
C         BOX REC RAW                                                   GENI 215
  120 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LT+3*J3                 GENI 216
      MASTER(LDATA+1)=LT+3*J4                                           GENI 217
      LDATA=LDATA+2                                                     GENI 218
      GOTO 360                                                          GENI 219
C         SPH                                                           GENI 220
  130 MASTER(LDATA)=(LT+3*J1)*I15+LSCAL+J2-1+I30                        GENI 221
      LDATA=LDATA+1                                                     GENI 222
      GOTO 360                                                          GENI 223
C         RCC                                                           GENI 224
  140 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1              GENI 225
      LDATA=LDATA+1                                                     GENI 226
      GOTO 360                                                          GENI 227
C         TRC TOR                                                       GENI 228
  150 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1              GENI 229
      MASTER(LDATA+1)=LSCAL+J4-1                                        GENI 230
      LDATA=LDATA+2                                                     GENI 231
      GOTO 360                                                          GENI 232
C         ELL                                                           GENI 233
  160 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1              GENI 234
      LDATA=LDATA+1                                                     GENI 235
      GOTO 360                                                          GENI 236
C         TEC                                                           GENI 237
  170 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LT+3*J3                 GENI 238
      MASTER(LDATA+1)=(LT+3*J4)*I30+(LSCAL+J5-1)*I15+LSCAL+J6-1         GENI 239
      MASTER(LDATA+2)=LSCAL+J7-1                                        GENI 240
      LDATA=LDATA+3                                                     GENI 241
```

```
      GOTO 360                                                          GENI 242
C                                                                       GENI 243
C           BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS                 GENI 244
  200 GOTO(201,220,202,201,203,202,201,230,204,203,240),ITYPE           GENI 245
  201 LE=12                                                             GENI 246
      GOTO 210                                                          GENI 247
  202 LE= 7                                                             GENI 248
      GOTO 210                                                          GENI 249
  203 LE= 8                                                             GENI 250
      GOTO 210                                                          GENI 251
  204 LE=13                                                             GENI 252
  210 WRITE (6,915)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(FX(J),J=1,6)  GENI 253
      READ(5,940)(FX(J),J=7,LE)                                         GENI 254
      WRITE (6,916)(FX(J),J=7,LE)                                       GENI 255
C           BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS                 GENI 256
      GOTO(290,300,300,290,285,270,290,300,260,250,300),ITYPE           GENI 257
C           SPH                                                         GENI 258
  220 WRITE (6,915)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(FX(J),J=1,4)  GENI 259
      GOTO 300                                                          GENI 260
C           ARB                 / R1 / V1 /      6 PER ARB              GENI 261
  230 WRITE (6,915)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(FX(J),J=1,6)  GENI 262
      CALL ALBERT(FX,LBOT,NDO,LS1)                                      GENI 263
      GOTO 360                                                          GENI 264
C           ARS                 /     / LDATA /                         GENI 265
  240 CALL ARIN(LBOT,LDATA,MASTER,ASTER,IWH)                            GENI 266
      GOTO 360                                                          GENI 267
C           TOR           CONVERT TO UNIT VECTOR                        GENI 268
  250 TT(1)=FX(4)                                                       GENI 269
      TT(2)=FX(5)                                                       GENI 270
      TT(3)=FX(6)                                                       GENI 271
      CALL UNIT(TT)                                                     GENI 272
      FX(4)=TT(1)                                                       GENI 273
      FX(5)=TT(2)                                                       GENI 274
      FX(6)=TT(3)                                                       GENI 275
      IF(FX(7).GE.FX(8))GOTO 280                                        GENI 276
      WRITE (6,943)                                                     GENI 277
      IERR=IERR+1                                                       GENI 278
      GOTO 280                                                          GENI 279
C           TEC                                                         GENI 280
  260 FX(15)=FX(13)                                                     GENI 281
      LE=15                                                             GENI 282
      TT1(1)=FX(7)                                                      GENI 283
      TT1(2)=FX(8)                                                      GENI 284
      TT1(3)=FX(9)                                                      GENI 285
      TT2(1)=FX(10)                                                     GENI 286
      TT2(2)=FX(11)                                                     GENI 287
      TT2(3)=FX(12)                                                     GENI 288
      IF(ABS(DOT(TT1,TT2)).LE.0.01) GOTO 265                            GENI 289
      WRITE (6,942)NN,TT1,TT2                                           GENI 290
      IERR=IERR+1                                                       GENI 291
C           SEMI MAJOR AXIS FX(13)                                      GENI 292
  265 FX(13)=SQRT(DOT(TT1,TT1))                                         GENI 293
      CALL UNIT(TT1)                                                    GENI 294
      FX(10)=TT1(1)                                                     GENI 295
      FX(11)=TT1(2)                                                     GENI 296
      FX(12)=TT1(3)                                                     GENI 297
C           SEMI MINOR AXIS FX(14)                                      GENI 298
      FX(14)=SQRT(DOT(TT2,TT2))                                         GENI 299
C           NORMAL HEIGHT VECTOR                                        GENI 300
      CALL CROSS(TT,TT1,TT2)                                            GENI 301
```

```
      CALL UNIT(TT)                                                     GENI 302
      HDN=FX(4)*TT(1)+FX(5)*TT(2)+FX(6)*TT(3)                           GENI 303
      IF(HDN)267,266,268                                                GENI 304
  266 WRITE(6,948)                                                      GENI 305
      IERR=IERR+1                                                       GENI 306
      GOTO 268                                                          GENI 307
  267 TT(1)=-TT(1)                                                      GENI 308
      TT(2)=-TT(2)                                                      GENI 309
      TT(3)=-TT(3)                                                      GENI 310
  268 FX(7)=TT(1)                                                       GENI 311
      FX(8)=TT(2)                                                       GENI 312
      FX(9)=TT(3)                                                       GENI 313
      GOTO 280                                                          GENI 314
C         ELL            COMPUTE FOCI                                   GENI 315
  270 IF(IC(4).EQ.IBL)GOTO 300                                          GENI 316
      ASQ=FX(4)*FX(4)+FX(5)*FX(5)+FX(6)*FX(6)                           GENI 317
      C=SQRT(ASQ-FX(7)*FX(7))                                           GENI 318
      A=SQRT(ASQ)                                                       GENI 319
      FX(7)=A+A                                                         GENI 320
C         X,Y,Z COMPONENTS OF FOCI                                      GENI 321
      CX=C*FX(4)/A                                                      GENI 322
      CY=C*FX(5)/A                                                      GENI 323
      CZ=C*FX(6)/A                                                      GENI 324
C         VERTEX + AND - X,Y,Z COMPONENTS GIVE THE 2 FOCI               GENI 325
      FX(4)=FX(1)+CX                                                    GENI 326
      FX(5)=FX(2)+CY                                                    GENI 327
      FX(6)=FX(3)+CZ                                                    GENI 328
      FX(1)=FX(1)-CX                                                    GENI 329
      FX(2)=FX(2)-CY                                                    GENI 330
      FX(3)=FX(3)-CZ                                                    GENI 331
C         PRINT NEW INPUT                                               GENI 332
  280 WRITE (6,915)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(FX(J),J=1,6)  GENI 333
      WRITE (6,916)(FX(J),J=7,LE)                                       GENI 334
      GOTO 300                                                          GENI 335
C         TRC            CHECK R1.NE.R2                                 GENI 336
  285 IF(FX(7).NE.FX(8))GOTO 300                                        GENI 337
      WRITE (6,944)                                                     GENI 338
      IERR=IERR+1                                                       GENI 339
      GOTO 300                                                          GENI 340
C         BOX RAW REC            CHECK IF VECTORS ARE PERPINDICULAR     GENI 341
  290 IF(ABS(FX(4)*FX(7)+FX(5)*FX(8)+FX(6)*FX(9)).LE.0.01)GOTO 291      GENI 342
      WRITE (6,942)NN,(FX(J),J=4,9)                                     GENI 343
      IERR=IERR+1                                                       GENI 344
  291 IF(ABS(FX(4)*FX(10)+FX(5)*FX(11)+FX(6)*FX(12)).LE.0.01)GOTO 292   GENI 345
      WRITE (6,942)NN,FX(4),FX(5),FX(6),FX(10),FX(11),FX(12)            GENI 346
      IERR=IERR+1                                                       GENI 347
  292 IF(ABS(FX(7)*FX(10)+FX(8)*FX(11)+FX(9)*FX(12)).LE.0.01)GOTO 300   GENI 348
      WRITE (6,942)NN,(FX(J),J=7,12)                                    GENI 349
      IERR=IERR+1                                                       GENI 350
C                                                                       GENI 351
C         BOX SPH RCC RLC TRC ELL RAW ARB TEC TOR ARS                   GENI 352
  300 GOTO(310,320,330,310,340,330,310,230,350,340,240),ITYPE           GENI 353
C         BOX REC RAW            / V1 / V2 /                            GENI 354
C                                / V2 / V3 /                            GENI 355
  310 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)  GENI 356
      MASTER(LDATA)=IWH*I15                                             GENI 357
      CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)  GENI 358
      MASTER(LDATA)=MASTER(LDATA)+IWH                                   GENI 359
      CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(8),FX(9),LBOT,LDATA,NDQ,LS1)  GENI 360
      MASTER(LDATA+1)=IWH*I15                                           GENI 361
```

```
      CALL SEE3(IWH,ASTER,MASTER,FX(10),FX(11),FX(12),                  GENI 362
     1  LBOT,LDATA,NDQ,LS1)                                             GENI 363
      MASTER(LDATA+1)=MASTER(LDATA+1)+IWH                               GENI 364
      LDATA=LDATA+2                                                     GENI 365
      GO TO 360                                                         GENI 366
C         SPH                    / V1 / R1 /                            GENI 367
  320 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)  GENI 368
      MASTER(LDATA)=IWH*I15                                             GENI 369
      LS1=1                                                             GENI 370
      CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(4),FX(4),LBOT,LDATA,NDQ,LS1)  GENI 371
      LS1=0                                                             GENI 372
      MASTER(LDATA)=MASTER(LDATA)+IWH                                   GENI 373
      LDATA=LDATA+1                                                     GENI 374
      GOTO 360                                                          GENI 375
C         RCC FLL                / V1 / V2 /                            GENI 376
C                                /    / R1 /                            GENI 377
  330 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)  GENI 378
      MASTER(LDATA)=IWH*I15                                             GENI 379
      CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)  GENI 380
      MASTER(LDATA)=MASTER(LDATA)+IWH                                   GENI 381
      LS1=1                                                             GENI 382
      CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(7),FX(7),LBOT,LDATA,NDQ,LS1)  GENI 383
      LS1=0                                                             GENI 384
      MASTER(LDATA+1)=IWH                                               GENI 385
      LDATA=LDATA+2                                                     GENI 386
      GO TO 360                                                         GENI 387
C         TRC TOR                / V1 / V2 /                            GENI 388
C                                / R1 / R2 /                            GENI 389
  340 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)  GENI 390
      MASTER(LDATA)=IWH*I15                                             GENI 391
      CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)  GENI 392
      MASTER(LDATA)=MASTER(LDATA)+IWH                                   GENI 393
      LS1=1                                                             GENI 394
      CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(7),FX(7),LBOT,LDATA,NDQ,LS1)  GENI 395
      MASTER(LDATA+1)=IWH*I15                                           GENI 396
      CALL SEE3(IWH,ASTER,MASTER,FX(8),FX(8),FX(8),LBOT,LDATA,NDQ,LS1)  GENI 397
      LS1=0                                                             GENI 398
      MASTER(LDATA+1)=MASTER(LDATA+1)+IWH                               GENI 399
      LDATA=LDATA+2                                                     GENI 400
      GO TO 360                                                         GENI 401
C         TEC                    / V1 / V2 /                            GENI 402
C                                / V3 / V4 /                            GENI 403
C                                / R1 / R2 /                            GENI 404
C                                /    / R3 /                            GENI 405
  350 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)  GENI 406
      MASTER(LDATA)=IWH*I15                                             GENI 407
      CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)  GENI 408
      MASTER(LDATA)=MASTER(LDATA)+IWH                                   GENI 409
      CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(8),FX(9),LBOT,LDATA,NDQ,LS1)  GENI 410
      MASTER(LDATA+1)=IWH*I15                                           GENI 411
      CALL SEE3(IWH,ASTER,MASTER,FX(10),FX(11),FX(12),                  GENI 412
     1  LBOT,LDATA,NDQ,LS1)                                             GENI 413
      MASTER(LDATA+1)=MASTER(LDATA+1)+IWH                               GENI 414
      LS1=1                                                             GENI 415
      CALL SEE3(IWH,ASTER,MASTER,FX(13),FX(13),FX(13),                  GENI 416
     1  LBOT,LDATA,NDQ,LS1)                                             GENI 417
      MASTER(LDATA+2)=IWH*I15                                           GENI 418
      CALL SEE3(IWH,ASTER,MASTER,FX(14),FX(14),FX(14),                  GENI 419
     1  LBOT,LDATA,NDQ,LS1)                                             GENI 420
      MASTER(LDATA+2)=MASTER(LDATA+2)+IWH                               GENI 421
```

```
      CALL SEE3(IWH,ASTER,MASTER,FX(15),FX(15),FX(15),
     1  LBOT,LDATA,NDQ,LS1)
      LS1=0
      MASTER(LDATA+3)=IWH
      LDATA=LDATA+4
C         CHECK IF ANY MORE ROOM FOR SOLID DATA
  360 IF(LDATA.LT.NDQ)GOTO 370
      WRITE (6,917)LDATA,LBOT,NDQ
      STOP
  370 CONTINUE
      WRITE (6,918)
      WRITE(6,947)ITY,NBOD
      WRITE (6,945)LBASE,LRPPD,LABUT,LBODY,LBOD,LDATA,LBOT,LSCAL,LTRIP,N
     ADQ
C
C     TRANSFER ASTER(LBOT - NDQ) TO ASTER(LDATA - LDATA+LSUB)
C
      LD=LDATA-1
      LSUB=LBOT-LD-1
      DO 375 I=LBOT,NDQ
      ASTER(LDATA)=ASTER(I)
      LDATA=LDATA+1
  375 CONTINUE
C         UNPACK POINTERS AND ADJUST FOR TRANSFER
      K=LBODY+3*(NRPP+NBODY)
      DO 390 I=K,LD
      CALL UN2(I,I1,I2)
      IF(I1.NE.0)I1=I1-LSUB
      IF(I2.NE.0)I2=I2-LSUB
      MASTER(I)=I1*I15+I2
  390 CONTINUE
C
C     REGION STORAGE
C         LREGD     / LOC BODY LIST / NUM OF BODIES /
C         LDATA     / OPERATOR      / BODY NUM      /
C
C
      WRITE (6,920)
      N=0
      J=0
      LREGD=LDATA
      LDATA=LDATA+NRMAX
      LREGL=LDATA
C
C     READ REGION
C
  400 READ(5,921)IR,(IA(I),IN(I),I=1,9)
C         CHECK VALIDITY OF REGION DATA
      DO 410 I=1,9
      IF(IABS(IN(I)).LE.NBODY+NRPP)GOTO 410
      WRITE (6,922)IR,I
      J=J+1
  410 CONTINUE
C
C     STORE REGION DATA
C
      IF(IR)440,420,421
  420 WRITE (6,923)(IA(I),IN(I),I=1,9)
      GOTO 430
  421 N=N+1
```

```
      WRITE (6,924)IR,(IA(I),IN(I),I=1,9)
      M=LREGD+N-1
      MASTER(M)=LDATA*I15
C         CHECK OPERATOR
  430 DO 435 I=1,9
      DO 431 K=1,8
      IF(IA(I).EQ.IAA(K))GOTO 432
  431 CONTINUE
      WRITE (6,925)IA(I),I
      STOP
  432 IA(I)=IAM(K)
      IF(IN(I))433,400,434
  433 IA(I)=4+IA(I)
      IN(I)=-IN(I)
  434 MASTER(LDATA)=IA(I)*I15+IN(I)
      LDATA=LDATA+1
      MASTER(M)=MASTER(M)+1
      IF(LDATA.LT.NDQ)GOTO 435
      WRITE (6,927)LDATA,NDQ
      STOP
  435 CONTINUE
      GOTO 400
C
C     END REGION READ
C
  440 IF(N.GE.NRMAX)GOTO 441
      WRITE (6,926)IR
      STOP
  441 IF(J.LE.0)GOTO 442
      WRITE (6,941)
      STOP
  442 WRITE (6,928)
C
C     IF(IRCHEK.NE.0)TEST REGION DATA
C         (ERROR IF POINT CAN BE IN MORE THAN 1 REGION)
C
      IF(IRCHEK.EQ.NO)GOTO 500
      WRITE (6,937)
      LL=0
      MIS=0
C
      DO 456 I=1,NRMAX
      JJ=I+1
      DO 455 J=JJ,NRMAX
      KRI=LREGD+I-1
      CALL UN2(KRI,LOCI,NUMI)
      KRJ=LREGD+J-1
      CALL UN2(KRJ,LOCJ,NUMJ)
      IF(NUMI.GE.NUMJ)GOTO 450
      IO=NUMI
      II=NUMI
      GOTO 451
  450 IO=NUMJ
      II=NUMI
      L=LOCI
      LOCI=LOCJ
      LOCJ=L
C
  451 DO 453 KO=1,IO
      KLK=LOCI+KO-1
```

```
      CALL UN2(KLK,IOPO,NBO)
      DO 452 KI=1,II
      KLK=LOCJ+KI-1
      CALL UN2(KLK,IOPI,NBI)
      IF(IOPO.NE.IOPI)GOTO 452
      IF(NBO .NE.NBI )GOTO 452
      MIS=MIS+1
      GOTO 453
  452 CONTINUE
  453 CONTINUE
      IF(MIS.NE.II)GOTO 454
      WRITE (6,929)J,I
      LL=LL+1
  454 MIS=0
  455 CONTINUE
      WRITE (6,930)I
  456 CONTINUE
      IF(LL.GT.0)STOP
      WRITE (6,937)
C
C     IS=+1    ENTERING TABLE STORED BY I15
C       WHICH REGIONS (J) A RAY MIGHT BE IN IF IT
C           ENTERS A GIVEN BODY (I)
C     IS=-1    LEAVING TABLE STORED BY 1
C       WHICH REGIONS (J) A RAY MIGHT GO INTO IF IT
C           LEAVES A GIVEN BODY (I)
C
  500 IS=-1
      NN=NBODY+NRPP
      LENLV=LDATA
      DO 590 MMM=1,2
      DO 580 I=1,NN
      M=LBODY+3*(I-1)
      IF(IS.GE.0)GO TO 510
      MASTER(M+1)=MASTER(M+1)+LDATA
      GO TO 520
  510 MASTER(M+1)=MASTER(M+1)+LDATA*I15
C
  520 DO 570 J=1,NRMAX
      ITEMP=LREGD+J-1
      CALL UN2(ITEMP,LOC,NC)
      CALL UN2(LOC,IOP,DUM)
      DO 560 N=1,NC
      MM=LOC+N-1
      CALL UN2(MM,IOPER,NUM)
      IF(NUM.NE.I)GOTO 560
      IF(IOP.EQ.1.OR.IOP.EQ.5)GOTO 540
      IF(IOPER.GT.4)GOTO 530
      IF(IS-1)560,550,560
  530 IF(IS+1)560,551,560
  540 IF(IS.LT.0)GOTO 551
  550 MASTER(M+2)=MASTER(M+2)+I15
      GO TO 552
  551 MASTER(M+2)=MASTER(M+2)+1
  552 MASTER(LDATA)=J
      LDATA=LDATA+1
      IF(LDATA.LT.NDQ)GOTO 570
      WRITE (6,931)LDATA,NDQ,MMM,I
      STOP
  560 CONTINUE
```

```
  570 CONTINUE                                                          GENI 602
  580 CONTINUE                                                          GENI 603
      WRITE (6,938)MMM                                                  GENI 604
      IS=IS+2                                                           GENI 605
  590 CONTINUE                                                          GENI 606
C         RIN STORAGE      ROUT STORAGE      G1 TEMP STORAGE            GENI 607
      L1=LDATA-1                                                        GENI 608
      NN=NRPP+NBODY                                                     GENI 609
      LRIN=LDATA+1                                                      GENI 610
      LROT=LRIN+NN                                                      GENI 611
      LIO=LROT+NN                                                       GENI 612
      LEGEOM=LIO+NN                                                     GENI 613
      WRITE (6,932)LEGEOM                                               GENI 614
      WRITE (6,919)LREGD,LREGL,LENLV,LRIN,LROT,LIO,LEGEOM               GENI 615
C                                                                       GENI 616
C     PRINT ENTERING AND LEAVING TABLE                                  GENI 617
C                                                                       GENI 618
      IF(IENTLV.EQ.NO)RETURN                                            GENI 619
      WRITE(6,946)                                                      GENI 620
      NBNR=NBODY+NRPP                                                   GENI 621
C                                                                       GENI 622
      DO 600 N=1,NBNR                                                   GENI 623
      LOC=LBODY+3*(N-1)                                                 GENI 624
      LOC=LOC+1                                                         GENI 625
      CALL UN2(LOC,LENT,LEAV)                                           GENI 626
      LOC=LOC+1                                                         GENI 627
      CALL UN2(LOC,NENT,NEAV)                                           GENI 628
      J1=LENT                                                           GENI 629
      J2=LENT+NENT-1                                                    GENI 630
      WRITE (6,933)N,J1,J2,(MASTER(K),K=J1,J2)                          GENI 631
      J1=LEAV                                                           GENI 632
      J2=LEAV+NEAV-1                                                    GENI 633
      WRITE (6,934)N,J1,J2,(MASTER(K),K=J1,J2)                          GENI 634
  600 CONTINUE                                                          GENI 635
C                                                                       GENI 636
C     MASTER-ASTER ARRAY OUTPUT                                         GENI 637
C                                                                       GENI 638
      IF(IPRIN.EQ.0)RETURN                                              GENI 639
      WRITE (6,935)                                                     GENI 640
C                                                                       GENI 641
      DO 620 K=LBASE,L1,3                                               GENI 642
      IK=K                                                              GENI 643
      IK2=K+2                                                           GENI 644
      M=0                                                               GENI 645
      DO 610 I=IK,IK2                                                   GENI 646
      M=M+1                                                             GENI 647
      CALL UN2(I,I1,I2)                                                 GENI 648
      NO1(M)=I1                                                         GENI 649
      NO2(M)=I2                                                         GENI 650
      O4(M)=ASTER(I)                                                    GENI 651
      NOO(M)=I                                                          GENI 652
  610 CONTINUE                                                          GENI 653
      WRITE (6,936)(NOO(L),NO1(L),NO2(L),O4(L),L=1,3)                   GENI 654
  620 CONTINUE                                                          GENI 655
      RETURN                                                            GENI 656
      END                                                               GENI 657
C                                                                       GENI 658
C                                                                       GENI 659
      SUBROUTINE RPPIN(LAR)                                             **** 15
      DIMENSION MASTER(30000),X(6)                                      RPPIN 2
```

```
      COMMON ASTER(30000)
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
     1    LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
      COMMON/RRPP/LRPPD,LABUT
      EQUIVALENCE(MASTER,ASTER)
C
  910 FORMAT(6F12.6)
  920 FORMAT(I8,17X,6F12.5)
  930 FORMAT(1H0,27HERROR IN DESCRIPTION OF RPP,I5,5X,10HMIN.GE.MAX)
  940 FORMAT(1H0,27HERROR IN DESCRIPTION OF RPP,7X,I10,10X,I10)
  950 FORMAT(10X,7HSURFACE,I5,8X,2E20.6)
C
C     N IS RPP NUMBER      J IS SURFACE NUMBER
C
C     MASTER-ASTER STORAGE FOR RPP
C
C     LBASE - RPP POINTERS           RESERVE  12 WORDS/RPP
C         /  I  /  J  /
C         /     /  K  /
C              I  (POINTER TO LIST OF ABUTING RPP'S)
C              J  (NUM OF RPP'S THAT ABUT THIS SURFACE)
C              K  (POINTER TO MIN OR MAX CORRESPONDING
C                  TO THIS SURFACE)
C
C     LRPPD - RPP DATA STARTING AT LBASE + 12 * NRPP
C         MIN OR MAX      K POINTS HERE
C
C     LABUT TO LBODY-1
C         LIST OF ABUTING RPP'S PACKED 1 OR 2/WORD
C         I POINTS HERE      /  1  /  2  /
C         J CONTAINS NUMBER IN LIST
C
      IERR=0
      N=1
      I=LBASE+12*NRPP
      LRPPD=I
   10 READ(5,910)(X(J),J=1,6)
      WRITE (6,920)N,(X(J),J=1,6)
      DO 20 J=1,6,2
      IF(X(J).LT.X(J+1))GOTO 20
      WRITE (6,930)N
      STOP
   20 CONTINUE
C
C     STORE MIN AND MAX BEGINNING AT LBASE + 12 * NRPP
C
      DO 33 J=1,6
      II=LBASE+12*NRPP
      L=LBASE+12*(N-1)+2*(J-1)
   30 IF(II.LT.I)GOTO 31
      ASTER(I)=X(J)
      MASTER(L+1)=I
      I=I+1
      GOTO 33
C         CHECK FOR DUPLICATION
   31 IF(X(J).EQ.ASTER(II))GOTO 32
      II=II+1
      GOTO 30
   32 MASTER(L+1)=II
```

```
   33 CONTINUE                                                          RPPIN 63
      IF(N.GE.NRPP)GOTO 40                                              RPPIN 64
      N=N+1                                                             RPPIN 65
      GOTO 10                                                           RPPIN 66
C                                                                       RPPIN 67
   40 LABUT=I                                                           RPPIN 68
      LAST=I-1                                                          RPPIN 69
      L=LAST                                                            RPPIN 70
C                                                                       RPPIN 71
C     SEARCH FOR ABUTING RPP'S TO SURFACE                               RPPIN 72
C                                                                       RPPIN 73
      DO 57 I=1,NRPP                                                    RPPIN 74
      DO 57 N=1,6                                                       RPPIN 75
      LL=0                                                              RPPIN 76
      M=1                                                               RPPIN 77
      K=LBASE+12*(I-1)+2*(N-1)                                          RPPIN 78
      MASTER(K)=(L+1)*I15+MASTER(K)                                     RPPIN 79
      NC=3*N-1-4*(N/2)                                                  RPPIN 80
      DO 56 J=1,NRPP                                                    RPPIN 81
      IF(I.EQ.J)GOTO 56                                                 RPPIN 82
      IF(S(I,N).NE.S(J,NC))GOTO 56                                      RPPIN 83
C                                                                       RPPIN 84
      DO 53 K=1,3                                                       RPPIN 85
      NN=N+NC                                                           RPPIN 86
      K41=4*K-1                                                         RPPIN 87
      IF(NN.EQ.K41)GOTO 53                                              RPPIN 88
      K2=2*K                                                            RPPIN 89
      K21=K2-1                                                          RPPIN 90
      IF(S(I,K21).GT.S(J,K21))GOTO 50                                   RPPIN 91
      IF(S(J,K21).LT.S(I,K2 ))GOTO 53                                   RPPIN 92
   50 IF(S(I,K21).GE.S(J,K2 ))GOTO 51                                   RPPIN 93
      IF(S(J,K2 ).LE.S(I,K2 ))GOTO 53                                   RPPIN 94
   51 IF(S(I,K2 ).GT.S(J,K2 ))GOTO 56                                   RPPIN 95
      IF(S(I,K21).LT.S(J,K21))GOTO 56                                   RPPIN 96
   53 CONTINUE                                                          RPPIN 97
      M=-M                                                              RPPIN 98
      IF(M.LT.0)GOTO 54                                                 RPPIN 99
      MASTER(L)=MASTER(L)+J                                             RPPIN100
      GOTO 55                                                           RPPIN101
   54 L=L+1                                                             RPPIN102
      MASTER(L)=J*I15                                                   RPPIN103
   55 LL=LL+1                                                           RPPIN104
   56 CONTINUE                                                          RPPIN105
      K=LBASE+12*(I-1)+2*(N-1)                                          RPPIN106
      MASTER(K)=MASTER(K)+LL                                            RPPIN107
   57 CONTINUE                                                          RPPIN108
C                                                                       RPPIN109
C     TEST VALIDITY OF RPP DATA                                         RPPIN110
C                                                                       RPPIN111
      IF(NRPP.LE.1)GOTO 63                                              RPPIN112
C                                                                       RPPIN113
      DO 62 J=1,6                                                       RPPIN114
      NRPP1=NRPP-1                                                      RPPIN115
      DO 61 I=1,NRPP1                                                   RPPIN116
      JJ=LBASE+12*(I-1)+2*(J-1)                                         RPPIN117
      CALL UN2(JJ,IDUM,I2)                                              RPPIN118
      I3=MASTER(JJ+1)                                                   RPPIN119
      IF(I2.NE.0)GOTO 61                                                RPPIN120
      II=I+1                                                            RPPIN121
      DO 60 K=II,NRPP                                                   RPPIN122
```

```
      KK=LBASE+12*(K-1)+2*(J-1)                                         RPPIN123
      CALL UN2(KK,IDUM,I5)                                              RPPIN124
      I6=MASTER(KK+1)                                                   RPPIN125
      IF(I5.NE.0)GOTO 60                                                RPPIN126
      IF(I3.EQ.I6)GOTO 60                                               RPPIN127
      IERR=IERR+1                                                       RPPIN128
      WRITE (6,940)I,K                                                  RPPIN129
      WRITE (6,950)J,ASTER(I3),ASTER(I6)                                RPPIN130
   60 CONTINUE                                                          RPPIN131
      GOTO 62                                                           RPPIN132
   61 CONTINUE                                                          RPPIN133
   62 CONTINUE                                                          RPPIN134
   63 LAR=L                                                             RPPIN135
      RETURN                                                            RPPIN136
      END                                                               RPPIN137
C                                                                       RPPIN138
C                                                                       RPPIN139
      SUBROUTINE ALBERT(FX,LBOT,NDQ,LS1)                                ****1.16
      DIMENSION MASTER(30000),IA(6,4),AA(8,3),F(4),FX(6)                ALBERT 2
      COMMON ASTER(30000)                                               ALBERT 3
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     ALBERT 4
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              ALBERT 5
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 ALBERT 6
      EQUIVALENCE(ASTER,MASTER)                                         ALBERT 7
C                                                                       ALBERT 8
  901 FORMAT(25X,6F12.5)                                                ALBERT 9
  902 FORMAT(10X,6(1X,4I1))                                             ALBERT10
  903 FORMAT(10X,6E10.3)                                                ALBERT11
  904 FORMAT(25X,6(4X,4I2))                                             ALBERT12
  905 FORMAT(1H0,15HUNDEFINED PLANE)                                    ALBERT13
  906 FORMAT(I5,10(E11.4))                                              ALBERT14
  907 FORMAT(1H0,26HFOUR POINTS NOT IN A PLANE)                         ALBERT15
  908 FORMAT(1H0,25HERROR IN SIDE DESCRIPTION)                          ALBERT16
  909 FORMAT(1H0,16HDEGENERATE PLANE,I5)                                ALBERT17
C                                                                       ALBERT18
      K=1                                                               ALBERT19
      DO 10 I=1,2                                                       ALBERT20
      DO 10 J=1,3                                                       ALBERT21
      AA(I,J)=FX(K)                                                     ALBERT22
      K=K+1                                                             ALBERT23
   10 CONTINUE                                                          ALBERT24
      READ(5,903)((AA(I,J),J=1,3),I=3,8)                                ALBERT25
      READ(5,902)((IA(I,J),J=1,4),I=1,6)                                ALBERT26
      WRITE  (6,901)((AA(I,J),J=1,3),I=3,8)                             ALBERT27
      WRITE  (6,904)((IA(I,J),J=1,4),I=1,6)                             ALBERT28
C                                                                       ALBERT29
      DO 70 I=1,6                                                       ALBERT30
      IX=IA(I,1)                                                        ALBERT31
      IY=IA(I,2)                                                        ALBERT32
      IZ=IA(I,3)                                                        ALBERT33
      X1=AA(IX,1)                                                       ALBERT34
      Y1=AA(IX,2)                                                       ALBERT35
      Z1=AA(IX,3)                                                       ALBERT36
      X2=AA(IY,1)                                                       ALBERT37
      Y2=AA(IY,2)                                                       ALBERT38
      Z2=AA(IY,3)                                                       ALBERT39
      X3=AA(IZ,1)                                                       ALBERT40
      Y3=AA(IZ,2)                                                       ALBERT41
      Z3=AA(IZ,3)                                                       ALBERT42
      D=X1*(Y2*Z3-Z2*Y3)-X2*(Y1*Z3-Z1*Y3)+X3*(Y1*Z2-Z1*Y2)              ALBERT43
```

```
      A=(-Y2*Z3+Z2*Y3+Y1*Z3-Z1*Y3-Y1*Z2+Z1*Y2)                          ALBERT44
      B=(X2*Z3-Z2*X3-X1*Z3+X3*Z1+X1*Z2-Z1*X2)                           ALBERT45
      C=(Y2*X3-X2*Y3-Y1*X3+X1*Y3+Y1*X2-X1*Y2)                           ALBERT46
      D12=(X1-X3)**2+(Y1-Y3)**2+(Z1-Z3)**2                              ALBERT47
      A2B2C2=A*A+B*B+C*C                                                ALBERT48
      IF(A2B2C2.NE.0.)GOTO 21                                           ALBERT49
      WRITE (6,909)I                                                    ALBERT50
      D=ABS(D)                                                          ALBERT51
      GOTO 61                                                           ALBERT52
   21 D1210=D12*1.0E-12                                                 ALBERT53
      IF(A2B2C2.GT.D1210)GOTO 22                                        ALBERT54
      WRITE (6,905)                                                     ALBERT55
      WRITE (6,906)I,A,B,C,D,D12                                        ALBERT56
      IERR=IERR+1                                                       ALBERT57
      GOTO 70                                                           ALBERT58
   22 S=SQRT(A2B2C2)                                                    ALBERT59
      WX=A/S                                                            ALBERT60
      WY=B/S                                                            ALBERT61
      WZ=C/S                                                            ALBERT62
      IC=IA(I,4)                                                        ALBERT63
      X4=AA(IC,1)                                                       ALBERT64
      Y4=AA(IC,2)                                                       ALBERT65
      Z4=AA(IC,3)                                                       ALBERT66
      D2=(-D-(A*X4)-(B*Y4)-(C*Z4))/((A*WX)+(B*WY)+(C*WZ))               ALBERT67
      D22=D2*D2                                                         ALBERT68
C         THE NEXT CARD BYPASSES THE 4TH POINT TEST                     ALBERT69
C     IF(D22.LE.0.01)GOTO 30 $ PRINT 907 $ IERR=IERR+1                  ALBERT70
      IF(D22.LE.1.01)GOTO 30                                            ALBERT71
      WRITE (6,907)                                                     ALBERT72
      IERR=IERR+1                                                       ALBERT73
      WRITE (6,906)I,A,B,C,D,D12,D2                                     ALBERT74
      GOTO 70                                                           ALBERT75
C                                                                       ALBERT76
   30 DO 31 K=1,4                                                       ALBERT77
      F(K)=0.                                                           ALBERT78
   31 CONTINUE                                                          ALBERT79
      L=1                                                               ALBERT80
      DO 32 J=1,8                                                       ALBERT81
      IF(J.EQ.IX.OR.J.EQ.IY.OR.J.EQ.IZ.OR.J.EQ.IC)GOTO 32               ALBERT82
      F(L)=A*AA(J,1)+B*AA(J,2)+C*AA(J,3)+D                              ALBERT83
      L=L+1                                                             ALBERT84
   32 CONTINUE                                                          ALBERT85
      M=0                                                               ALBERT86
      N=0                                                               ALBERT87
      J=0                                                               ALBERT88
C                                                                       ALBERT89
      DO 44 L=1,4                                                       ALBERT90
      IF(ABS(F(L)).LE.1.0E-6)GOTO 42                                    ALBERT91
      IF(F(L))41,42,43                                                  ALBERT92
   41 M=M+1                                                             ALBERT93
      GOTO 44                                                           ALBERT94
   42 N=N+1                                                             ALBERT95
      GOTO 44                                                           ALBERT96
   43 J=J+1                                                             ALBERT97
   44 CONTINUE                                                          ALBERT98
C                                                                       ALBERT99
      IF(N.EQ.0)GOTO 51                                                 ALBER100
      IF(M+N.EQ.4)GOTO 60                                               ALBER101
      IF(J+N.EQ.4)GOTO 61                                               ALBER102
      GOTO 52                                                           ALBER103
```

```
   51 IF(M.EQ.4)GOTO 60
      IF(J.EQ.4)GOTO 61                                                 ALBER104
   52 WRITE (6,908)                                                     ALBER105
      WRITE (6,906)I,A,B,C,D,D12,D2,(F(L),L=1,4)                        ALBER106
      IERR=IERR+1                                                       ALBER107
      GOTO 70                                                           ALBER108
C                                                                       ALBER109
   60 A=-A                                                              ALBER110
      B=-B                                                              ALBER111
      C=-C                                                              ALBER112
      D=-D                                                              ALBER113
   61 CALL SEE3(IWH,ASTER,MASTER,A,B,C,LBOT,LDATA,NDQ,LS1)              ALBER114
      MASTER(LDATA)=IWH                                                 ALBER115
      LS1=1                                                             ALBER116
      CALL SEE3(IWH,ASTER,MASTER,D,D,D,LBOT,LDATA,NDQ,LS1)              ALBER117
      LS1=0                                                             ALBER118
      MASTER(LDATA)=MASTER(LDATA)+IWH*I15                               ALBER119
      LDATA=LDATA+1                                                     ALBER120
   70 CONTINUE                                                          ALBER121
      RETURN                                                            ALBER122
      END                                                               ALBER123
C                                                                       ALBER124
C                                                                       ALBER125
      SUBROUTINE ARIN(LBOT,LDATA,MASTER,ASTER,IWH)                      ALBER126
      DIMENSION MASTER(30000),ASTER(30000)                              ****  17
      COMMON/UNCLE/NN,IC(4)                                             ARIN   2
C                                                                       ARIN   3
C         SEE ARS SUBROUTINE FOR STORAGE IN MASTER-ASTER ARRAY          ARIN   4
C                                                                       ARIN   5
  901 FORMAT(10X,2I10)                                                  ARIN   6
  902 FORMAT(10X,6E10.3)                                                ARIN   7
  903 FORMAT(I8,1X,3A1,2X,3HARS,2X,A4,6X,19HNUMBER OF CURVES IS,I6,     ARIN   8
     1    5X,29HNUMBER OF POINTS PER CURVE IS,I6/)                      ARIN   9
  904 FORMAT(25X,6F12.5)                                                ARIN  10
C                                                                       ARIN  11
C     MAX = NUM OF CURVES                                               ARIN  12
C     NAX = NUM OF POINTS/CURVE                                         ARIN  13
C                                                                       ARIN  14
      READ(5,901)MAX,NAX                                                ARIN  15
      WRITE (6,903)NN,(IC(I),I=1,4),MAX,NAX                             ARIN  16
      LBOT=LBOT-4*MAX*NAX-92                                            ARIN  17
      IWH=LBOT                                                          ARIN  18
      MASTER(LDATA)=IWH                                                 ARIN  19
      LDATA=LDATA+1                                                     ARIN  20
C                                                                       ARIN  21
      DO 50 M=1,MAX                                                     ARIN  22
      L1=LBOT+92+4*NAX*(M-1)                                            ARIN  23
      L2=L1+4*NAX-1                                                     ARIN  24
      READ(5,902)(ASTER(L),ASTER(L+1),ASTER(L+2),L=L1,L2,4)             ARIN  25
      WRITE (6,904)(ASTER(L),ASTER(L+1),ASTER(L+2),L=L1,L2,4)           ARIN  26
      WRITE (6,904)                                                     ARIN  27
   50 CONTINUE                                                          ARIN  28
      MASTER(LBOT)=0                                                    ARIN  29
      MASTER(LBOT+1)=MAX                                                ARIN  30
      MASTER(LBOT+2)=NAX                                                ARIN  31
      RETURN                                                            ARIN  32
      END                                                               ARIN  33
C                                                                       ARIN  34
C                                                                       ARIN  35
      SUBROUTINE SEE3(IWH,ASTER,MASTER,FX,FXX,FXXX,LBOT,LDATA,NDQ,LS1)  ARIN  36
                                                                        ****  18
```

```
      DIMENSION ASTER(30000),MASTER(30000)                              SEE3   2
C                                                                       SEE3   3
C     STORES TRIPLETS AND SCALARS IN MASTER-ASTER ARRAY                 SEE3   4
C                                                                       SEE3   5
      IF(LS1.NE.0)GOTO 50                                               SEE3   6
C         TRIPLETS                                                      SEE3   7
      IF(LBOT.GT.NDQ)GOTO 20                                            SEE3   8
      NDQ2=NDQ-2                                                        SEE3   9
      DO 10 I=LBOT,NDQ2                                                 SEE3  10
      IF(ASTER(I).NE.FX)GOTO 10                                         SEE3  11
      IF(ASTER(I+1).NE.FXX)GOTO 10                                      SEE3  12
      IF(ASTER(I+2).NE.FXXX)GOTO 10                                     SEE3  13
      IWH=I                                                             SEE3  14
      RETURN                                                            SEE3  15
   10 CONTINUE                                                          SEE3  16
   20 ASTER(LBOT-1)=FXXX                                                SEE3  17
      ASTER(LBOT-2)=FXX                                                 SEE3  18
      ASTER(LBOT-3)=FX                                                  SEE3  19
      LBOT=LBOT-3                                                       SEE3  20
      IWH=LBOT                                                          SEE3  21
      IF(LBOT.LE.LDATA)WRITE (6,30)LBOT,LDATA                           SEE3  22
      RETURN                                                            SEE3  23
   30 FORMAT(1H0,22HMEMORY OVERLAP IN SEE3,5X,5HLBOT=,I10,              SEE3  24
     1  5X,6HLDATA=,I10)                                                SEE3  25
C                                                                       SEE3  26
C         SCALARS                                                       SEE3  27
   50 DO 60 I=LBOT,NDQ                                                  SEE3  28
      IF(ASTER(I).NE.FX)GOTO 60                                         SEE3  29
      IWH=I                                                             SEE3  30
      RETURN                                                            SEE3  31
   60 CONTINUE                                                          SEE3  32
      ASTER(LBOT-1)=FX                                                  SEE3  33
      LBOT=LBOT-1                                                       SEE3  34
      IWH=LBOT                                                          SEE3  35
      RETURN                                                            SEE3  36
      END                                                               SEE3  37
C                                                                       SEE3  38
C                                                                       SEE3  39
      FUNCTION S(I,N)                                                   ****  19
      DIMENSION MASTER(30000)                                           S      2
      COMMON ASTER(30000)                                               S      3
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 S      4
      EQUIVALENCE(MASTER,ASTER)                                         S      5
C                                                                       S      6
C     S RETRIEVES COORDINATES OF ANY OF THE 6 SIDES OF AN RPP           S      7
C     I IS RPP NUMBER      N IS SURFACE NUMBER                          S      8
C                                                                       S      9
      L=LBASE+12*(I-1)+2*(N-1)                                          S     10
      LL=MASTER(L+1)                                                    S     11
      S=ASTER(LL)                                                       S     12
      RETURN                                                            S     13
      END                                                               S     14
C                                                                       S     15
C                                                                       S     16
      SUBROUTINE CONVRT(FX,IX,LE)                                       ****  20
      DIMENSION FX(6),IX(8)                                             CONVRT 2
C         LE NUMBER OF REFERENCES TO SCALARS AND TRIPLETS               CONVRT 3
C         INTEGRAL PART OF FX CONVERTED TO FIXED POINT NUM IN IX(I)     CONVRT 4
C         FRACTIONAL PART OF FX CONVERTED TO FIXED POINT NUM IN IX(I1)  CONVRT 5
      NFX=(LE+1)/2                                                      CONVRT 6
```

```
      DO 10 IFX=1,NFX                                                   CONVRT 7
      I1=2*IFX                                                          CONVRT 8
      I=I1-1                                                            CONVRT 9
      IX(I)=FX(IFX)+.000001                                             CONVRT10
      X=IX(I)                                                           CONVRT11
      IX(I1)=(FX(IFX)-X)*100000.+.00001                                 CONVRT12
   10 CONTINUE                                                          CONVRT13
      RETURN                                                            CONVRT14
      END                                                               CONVRT15
C                                                                       CONVRT16
C                                                                       CONVRT17
      SUBROUTINE GRID                                                   ****  21
      DIMENSION WP(3)                                                   GRID   2
      COMMON/PAREM/XB(3),WB(3),IR                                       GRID   3
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 GRID   4
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     GRID   5
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              GRID   6
      COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,         GRID   7
     1  ITR(200),CA,CE,SA,SE                                            GRID   8
      COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),              GRID   9
     1  TRAVEL,SN,V,H,IVIH                                              GRID  10
      COMMON/WALT/LIRFO,NG1ERR                                          GRID  11
      COMMON/HOYT/VREF,HREF                                             GRID  12
      COMMON/CELL/CELSIZ                                                GRID  13
      COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES     GRID  14
C                                                                       GRID  15
  901 FORMAT(8I10)                                                      GRID  16
  902 FORMAT(6E12.8)                                                    GRID  17
  903 FORMAT(1H0,2HNX,I5,5X,2HNY,I5,5X,7HIRSTART,I5,5X,4HIENC,I5,5X,    GRID  18
     1    6HNSTART,I6,5X,4HNEND,I6,5X,9HCELL SIZE,F7.2//                GRID  19
     2    17H DATUM LINE AT Z=,F10.3,27H WITH RESPECT TO THE ORIGIN/    GRID  20
     3    17H GROUND IS  AT Z=,F10.3,27H WITH RESPECT TO THE ORIGIN/    GRID  21
     4    17H XSHIFT IS  AT X=,F10.3,27H WITH RESPECT TO THE ORIGIN/    GRID  22
     5    17H YSHIFT IS  AT Y=,F10.3,27H WITH RESPECT TO THE ORIGIN/)   GRID  23
  904 FORMAT(1H ,7HAZIMUTH,F12.5,5X,9HELEVATION,F12.5,5X,               GRID  24
     1    13HBACK OFF DIST,F12.5)                                       GRID  25
  905 FORMAT(2E20.8,4E10.3)                                             GRID  26
  906 FORMAT(52H0THIS RAY WAS SUPPRESSED BECAUSE IT WAS BELOW GROUND)   GRID  27
  907 FORMAT(1H0,I5,15H  CELLS SKIPPED)                                 GRID  28
C                                                                       GRID  29
      READ (5,901)NX,NY,IRSTRT, IENC,NG1ERR,NSTART,NEND                 GRID  30
      READ (5,902)A,E,ENGTH,ZSHIFT,GROUND                               GRID  31
      READ (5,902)XSHIFT,YSHIFT,CELSIZ                                  GRID  32
      IF(IRSTRT .LE.0)IRSTRT=1                                          GRID  33
      IF(CELSIZ .LE.0.)CELSIZ=4.                                        GRID  34
      IF(NSTART.LE.0)NSTART=1                                           GRID  35
      IF(NEND.LE.NSTART)NEND=NX*NY                                      GRID  36
      IF(NG1ERR.LE.0)NG1ERR=25                                          GRID  37
C                                                                       GRID  38
      WRITE (6,903)NX,NY,IRSTRT, IENC,NSTART,NEND,CELSIZ,               GRID  39
     1    ZSHIFT,GROUND,XSHIFT,YSHIFT                                   GRID  40
      IF(IWOT.EQ.IYES)WRITE(1,905)A,E,XSHIFT,YSHIFT,ZSHIFT,CELSIZ       GRID  41
      WRITE (6,904)A,E,ENGTH                                            GRID  42
      RADIAN=.017453292519943                                           GRID  43
      AR=A*RADIAN                                                       GRID  44
      ER=E*RADIAN                                                       GRID  45
      WRITE (6,904)AR,ER,ENGTH                                          GRID  46
      SA=SIN(AR)                                                        GRID  47
      CA=COS(AR)                                                        GRID  48
      SE=SIN(ER)                                                        GRID  49
```

```
      CE=COS(ER)                                                        GRID  50
C                                                                       GRID  51
C      PROCESS KL CELLS IN GRID                                         GRID  52
C                                                                       GRID  53
      DO 40 KK=NSTART,NEND                                              GRID  54
      WB(1)=-CE*CA                                                      GRID  55
      WB(2)=-CE*SA                                                      GRID  56
      WB(3)=-SE                                                         GRID  57
      II=((KK-1)/NX)+1                                                  GRID  58
      J=KK-(II-1)*NX                                                    GRID  59
C         COMPUTE COORDINATES OF GRID CELL IN GRID PLANE                GRID  60
      CELL2=.5*CELSIZ                                                   GRID  61
      V=FLOAT((NY/2)-II)*CELSIZ +CELL2                                  GRID  62
      VREF=V+CELL2                                                      GRID  63
      H=FLOAT((NX/2)- J)*CELSIZ +CELL2                                  GRID  64
      HREF=H+CELL2                                                      GRID  65
      IV=RAN(-1)*10.                                                    GRID  66
      IH=RAN(-1)*10.                                                    GRID  67
      IVIH=10*IH+IV                                                     GRID  68
C         COMPUTE H,V AT RANDOM POINT IN GRID CELL                      GRID  69
      V=V+CELSIZ *FLOAT(IV)/10.+CELSIZ /20.                             GRID  70
      H=H+CELSIZ *FLOAT(IH)/10.+CELSIZ /20.                             GRID  71
C         X,Y,Z IN COORDINATE SYSTEM OF VEHICLE                         GRID  72
      XBS(1)=XSHIFT-V*CA*SE-H*SA                                        GRID  73
      XBS(2)=YSHIFT-V*SA*SE+H*CA                                        GRID  74
      XBS(3)=ZSHIFT+V*CE                                                GRID  75
      CALL TROPIC(WP)                                                   GRID  76
      XBS(1)=XBS(1)+WP(1)*1.0E-4                                        GRID  77
      XBS(2)=XBS(2)+WP(2)*1.0E-4                                        GRID  78
      XBS(3)=XBS(3)+WP(3)*1.0E-4                                        GRID  79
      XB(1)=XBS(1)-ENGTH*WB(1)                                          GRID  80
      XB(2)=XBS(2)-ENGTH*WB(2)                                          GRID  81
      XB(3)=XBS(3)-ENGTH*WB(3)                                          GRID  82
      IF(XB(3).GT.GROUND)GOTO 10                                        GRID  83
      IF(ITAPE8.EQ.IYES)WRITE (6,906)                                   GRID  84
      GOTO 40                                                           GRID  85
   10 DO 20 KK1=1,3                                                     GRID  86
      XS(KK1)=XB(KK1)                                                   GRID  87
      WS(KK1)=WB(KK1)                                                   GRID  88
   20 CONTINUE                                                          GRID  89
      CALL TRACK                                                        GRID  90
      IF(IERR.GE.NGIERR)RETURN                                          GRID  91
      IF(IRAYSK.EQ.NO)GOTO 40                                           GRID  92
      MSHIFT=RAN(-1)*25.                                                GRID  93
      WRITE (6,907)MSHIFT                                               GRID  94
      KK=KK+MSHIFT                                                      GRID  95
   40 CONTINUE                                                          GRID  96
      RETURN                                                            GRID  97
      END                                                               GRID  98
C                                                                       GRID  99
C                                                                       GRID 100
C                                                                       GRID 101
      SUBROUTINE TRACK                                                  ****  22
      DIMENSION XP(3),ERROR(2)                                          TRACK  2
      COMMON/PAREN/XB(3),WB(3),IR                                       TRACK  3
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 TRACK  4
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     TRACK  5
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              TRACK  6
      COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,         TRACK  7
     1  ITR(200),CA,CE,SA,SE                                            TRACK  8
```

```
      COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),TRAVEL,
     1  SN,V,H,IVIH
      COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES
      COMMON/WALT/LIRFO,NGIERR
      COMMON/HOYT/VREF,HREF
      COMMON/LSU/LSURF
      COMMON/CELL/CELSIZ
      COMMON/ERR/IERRO
C
  901 FORMAT(F6.1,1X,F6.1,1X,I2,1X,F7.2,1X,F7.2,4(1X,I1),I3,1X,2I3,
     1    1X,F8.3,1X,F8.3)
  902 FORMAT(2(I4,F7.2,F7.2,F6.1,I3,F7.2),1X,2I3,1X,I1,I1,I2,4X,A6)
  903 FORMAT(31H NUMBER OF INTERSECTIONS.GT.200)
  904 FORMAT(//)
  905 FORMAT(1H0,16H0 ITEM IN CELL (,I4,1H,,I4,1H),5X,
     1    2HH=,F6.1,5X,2HV=,F6.1)
C
      ERROR(2)= 6H0 ITEM
      DATA ERROR(1),ERROR(2)/4H    ,4HITEM/
      I12=4096
      NASC=-1
      IR=IRSTRT
      L=1
      KHIT=0
      JCNT=0
      MSKRT=0
      MTARG=1
      MARMR=0
      MVOL =0
C
      DO 10 I=1,200
      ITR(I)=0
      TR(I)=0.
   10 CONTINUE
C
C     S1 IS DISTANCE THRU REGION IR
C     IRPRIM IS NEW REGION NUMBER
C     XP IS POINT OF CONTACT
C
   20 CALL G1(S1,IRPRIM,XP)
      IF(IRPRIM.LT.0)RETURN
      TR(L)=S1
      KLSURF=LSURF+7
      LOC=LIRFO+IR-1
      CALL UN2(LOC,DUM,IDENT)
      IDENT=IDENT-1
C         / SURFACE NUM / BODY NUM / NEXT REGION /
      ITR(L)=(KLSURF*I12+NASC)*I12+IRPRIM
      IF(NASC.LE.NRPP)IRPRIM=0
      IF(IRPRIM.EQ.0)GOTO 100
      IR=IRPRIM
      KHIT=KHIT+1
      IF(L.GT.1)GOTO 40
      SUM=0.
      DO 30 I=1,3
      SUM=SUM+WS(I)*XP(I)
   30 CONTINUE
      D1=-SUM
      GOTO 60
C
```

```
C      CHECK IDENT CODE     0 NONE                                          TRACK 69
C          10=SKIRT     20=ARMOR     30=TARGET                             TRACK 70
C          SPACE CODES     1 EXTERIOR VOLUME                               TRACK 71
C              -1,2-9,11-19,21-29,.......,91-99 INTERIOR VOLUME            TRACK 72
C                                                                          TRACK 73
   40 IF(IDENT.EQ.0)GOTO 60                                                TRACK 74
      IF(IDENT-(IDENT/10)*10.EQ.0)GOTO 50                                  TRACK 75
      KHIT=KHIT-1                                                          TRACK 76
      IF(IDENT.NE.1)MVOL=1                                                 TRACK 77
      GOTO 50                                                              TRACK 78
C                                                                          TRACK 79
   50 IF(IDENT.EQ.20)MARMR=1                                               TRACK 80
      IF(IDENT.EQ.30)MTARG=1                                               TRACK 81
      IF(IDENT.EQ.10)MSKRT=1                                               TRACK 82
   60 L=L+1                                                                TRACK 83
      IF(L.LE.200)GOTO 20                                                  TRACK 84
      WRITE (6,903)                                                        TRACK 85
      STOP                                                                 TRACK 86
C                                                                          TRACK 87
C      END OF RAY     PRINT RESULTS                                        TRACK 88
C                                                                          TRACK 89
  100 IF(L.EQ.1)RETURN                                                     TRACK 90
      IF(IIAPES.EQ.NO.AND.IWOT.EQ.NO)RETURN                                TRACK 91
      D2=XDIST(XBS,XP)-S1                                                  TRACK 92
      D2=-D2                                                               TRACK 93
      IF(KHIT.GT.0)GOTO 105                                                TRACK 94
      KHIT=KHIT+1                                                          TRACK 95
      MTARG=0                                                              TRACK 96
  105 KHIT=KHIT-1                                                          TRACK 97
      IH=ABS(H/CELSIZ )+.5                                                 TRACK 98
      IF(H.LT.0.)IH=-IH                                                    TRACK 99
      IV=ABS(V/CELSIZ )+.5                                                 TRACK100
      IF(V.LT.0.)IV=-IV                                                    TRACK101
C         PRINT CARD NUM 1                                                 TRACK102
      IF(IIAPES.EQ.NO)GOTO 110                                             TRACK103
      WRITE (6,904)                                                        TRACK104
      WRITE (6,901)HREF,VREF,IVIH,D1,D2,MSKRT,MTARG,MARMR,MVOL,            TRACK105
     1    KHIT,IH,IV,H,V                                                   TRACK106
  110 IF(IWOT.EQ.IYES)WRITE(1,901)HREF,VREF,IVIH,D1,D2,MSKRT,MTARG,        TRACK107
     1    MARMR,MVOL,KHIT,IH,IV,H,V                                        TRACK108
C                                                                          TRACK109
C       PROCESS COMPONENT CARDS                                            TRACK110
C                                                                          TRACK111
      LMAX=L                                                               TRACK112
      L=0                                                                  TRACK113
      TRAVEL=TR(1)                                                         TRACK114
C                                                                          TRACK115
C     NIR      REGION IDENTIFICATION(VEHICLE COMPONENT)                    TRACK116
C     SIN      LINE-OF-SIGHT DISTANCE                                      TRACK117
C     ANGLE    OBLIQUITY ANGLE                                             TRACK118
C     SN       NORMAL DISTANCE THRU REGION                                 TRACK119
C     NTYPE    TYPE OF SPACE AFTER NIR(NONE=0,END RAY=9)                   TRACK120
C     SSPACE   LINE-OF-SIGHT DISTANCE THRU SPACE                           TRACK121
C                                                                          TRACK122
      DO 200 KIK=1,LMAX,2                                                  TRACK123
      JERRO=1                                                              TRACK124
      L=L+1                                                                TRACK125
      IF(L.GE.LMAX)RETURN                                                  TRACK126
      CALL CALC                                                            TRACK127
      IF(NIR.NE.0)GOTO 113                                                 TRACK128
```

```
      JERRO=2                                                           TRACK129
      IERRO=IERRO+1                                                     TRACK130
  113 IF(SSPACE.NE.0.)JCNT=JCNT+1                                       TRACK131
      NIR1=NIR                                                          TRACK132
      SIN1=SIN                                                          TRACK133
      ANGLE1=ANGLE                                                      TRACK134
      SN1=SN                                                            TRACK135
      NTYPE1=NTYPE                                                      TRACK136
      SSPACE1=SSPACE                                                    TRACK137
C         SECOND HALF OF CARD                                           TRACK138
      L=L+1                                                             TRACK139
      IF(L.LT.LMAX)GOTO 115                                             TRACK140
      NIR=0                                                             TRACK141
      SIN=0.                                                            TRACK142
      ANGLE=0.                                                          TRACK143
      SN=0.                                                             TRACK144
      NTYPE=0                                                           TRACK145
      SSPACE=0.                                                         TRACK146
      GOTO 120                                                          TRACK147
  115 CALL CALC                                                         TRACK148
      IF(NIR.NE.0)GOTO 117                                              TRACK149
      JERRO=2                                                           TRACK150
      IERRO=IERRO+1                                                     TRACK151
  117 IF(SSPACE.EQ.0.)GOTO 130                                          TRACK152
  120 JCNT=JCNT+1                                                       TRACK153
  130 I1=0                                                              TRACK154
      I2=0                                                              TRACK155
      N=L-JCNT                                                          TRACK156
C                                                                       TRACK157
C     TRACK FLAG    501 IS TRACK EDGE    502 IS TRACK FACE              TRACK158
C     10 IN. NORMAL THICKNESS IS CUTOFF                                 TRACK159
C                                                                       TRACK160
      IF(NIR1.NE.501)GOTO 140                                           TRACK161
      IF(SN1.LT.10.)NIR1=502                                            TRACK162
  140 IF(NIR.NE.501)GOTO 150                                            TRACK163
      IF(SN .LT.10.)NIR=502                                             TRACK164
C                                                                       TRACK165
C     PRINT COMPONENT CARD                                              TRACK166
C                                                                       TRACK167
  150 IF(IWOT.EQ.IYES)WRITE(1,902)NIR1,SIN1,SN1,ANGLE1,NTYPE1,SSPACE1,  TRACK168
     1    NIR,SIN,SN,ANGLE,NTYPE,SSPACE,IH,IV,I1,I2,N                   TRACK169
      IF(ITAPE8.EQ.IYES)WRITE(6,902)NIR1,SIN1,SN1,ANGLE1,NTYPE1,SSPACE1,TRACK170
     1    NIR,SIN,SN,ANGLE,NTYPE,SSPACE,IH,IV,I1,I2,N,ERROR(JERRO)      TRACK171
      IF(ITAPE8.EQ.NO.AND.JERRO.EQ.2)WRITE(6,905)IH,IV,HREF,VREF        TRACK172
C                                                                       TRACK173
      IF(L.GE.LMAX)RETURN                                               TRACK174
      IF(NTYPE .EQ.9)RETURN                                             TRACK175
  200 CONTINUE                                                          TRACK176
      RETURN                                                            TRACK177
      END                                                               TRACK178
C                                                                       TRACK179
C                                                                       TRACK180
      SUBROUTINE CALC                                                   ****  23
      DIMENSION MASTER(30000),XP(3),TEMP(3),TEMP1(3),TEM(3),TEM1(3),    CALC   2
     1  XMID(3),IEMP(4),WN(3),WI(3),WA(3),XI(3),AUN(3),HF(3),           CALC   3
     2  VF(3),Q(3),DELTA(3),ARSTP(3)                                    CALC   4
      COMMON ASTER(30000)                                               CALC   5
      COMMON/PAREM/XB(3),WB(3),IR                                       CALC   6
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 CALC   7
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     CALC   8
```

```
     1  LDATA,LCIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
      COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,
     1  IP2(200),CA,CE,SA,SE
      COMMON/WALE/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),TRAVEL,
     1  XN,V,H,IVIR
      COMMON/WALT/LIREG,NGIERR
      EQUIVALENCE(MASTER,ASTER)
      REAL NF(3)
C
  901 FORMAT(1H0,15HTHATS ALL FOLKS//)
  902 FORMAT(1H0,17HBAD ITYPE IN CALC,5X,6HITYPE=,I5,4HNBO=,I5/
     1  16H RETURN TO TRACK//)
  903 FORMAT(1H0,23HARS DID NOT FIND NORMAL)
  904 FORMAT(//5H NORM/5H NIR=,I10,5X,6HITYPE=,I10,5X,4HNBO=,I10,5X,
     1  6HLSURF=,I10/4H WB=,3E20.10/4H WS=,3E20.10/4H XP=,3E20.8/
     2  4H XB=,3E20.10/4H XI=,3E20.10/6H XNOS=,3E20.10)
  905 FORMAT(5H ERROR IN CALC    A TRC HAS R1 = R2 )
  906 FORMAT(42H ERROR IN CALC    BAD LSURF FOR BOX OR RAW )
C
C         / SURFACE NUM / BODY NUM / NEXT REGION /
C
      CALL OPENK(L,LSURF,NBO,NIR)
      IF(IN.GT.0)GOTO 10
      WRITE (6,901)
      RETURN
C
C     TRAVEL    LINE-OF-SIGHT DIST TO THIS REGION
C     XS        STARTING POINT  (XS=XB)
C     SIN       LINE-OF-SIGHT DIST THRU THIS REGION
C
   10 SIN=TR(L+1)
      DO 20 I=1,3
      XI(I)=XS(I)+TRAVEL*WS(I)
   20 CONTINUE
      TRAVEL=TRAVEL+SIN
      LSURF=LSURF-7
C
      XNOS=1.
      IF(LSURF.LT.0)XNOS=-1.
      LOC=LBODY+3*(NBO-1)
      CALL UN2(LOC,ITYPE,LDATA)
      LSURF=IABS(LSURF)
      ITYPE=ITYPE+1
      IF(ITYPE.GE.1.AND.ITYPE.LE.12)GOTO 30
      WRITE (6,902)ITYPE,NBO
      RETURN
C
C     COMPUTE NORMAL DIST AND OBLIQUITY ANGLE
C
C         RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS
   30 GOTO(50,100,150,200,200,300,350,400,450,500,550,600),ITYPE
C
C     CHECK FOR SPACE CODES    IDENT = -1,1-9,11-19,21-29,....,91-99
C
   40 CALL OPENK(L+1,DUM,DUM,NEXREG)
      ISPOT=LIREG+NEXREG-1
      CALL UN2(ISPOT,DUM,IDENT)
      ISPOT=LIREG+NIR-1
      CALL UN2(ISPOT,NIR,DUM)
      IDENT=IDENT-1
```

```
C          CHECK FOR SPACE CODES    IDENT = -1,1-9,11-19,21-29,.....
      IF(IDENT-(IDENT/10)*10.NE.0)GOTO 41
      NTYPE=0
      SSPACE=0.
      RETURN
   41 L=L+1
      IF(L+1.LT.LMAX)GOTO 42
      IDENT=9
      SSPACE=1.0E-4
      NTYPE=IDENT
      RETURN
   42 NTYPE=IDENT
      SSPACE=TR(L+1)
      TRAVEL=TRAVEL+SSPACE
      RETURN
C
C     RPP
C
   50 IF(LSURF-2)52,53,54
   52 XNOS=-XNOS
   53 I=1
      GOTO 60
   54 IF(LSURF-4)55,56,57
   55 XNOS=-XNOS
   56 I=3
      GOTO 60
   57 IF(LSURF.GE.6)GOTO 59
      XNOS=-XNOS
   59 I=5
   60 LKK=LBASF+2*I+1
      LV1=MASTER(LKK)
      LKK=LKK+2
      LV2=MASTER(LKK)
      DO 62 J=1,3
      M=J-1
      IJK=M+LV1
      TEMP(J)=ASTER(IJK)
      IJK=M+LV2
      TEMP1(J)=ASTER(IJK)
   62 CONTINUE
      CALL DCOSP(TEMP,TEMP1,WB)
      DO 63 J=1,3
      WB(J)=XNOS*WB(J)
   63 CONTINUE
      GOTO 1000
C
C     BOX
C
  100 CONTINUE
      KCOM=LSURF-(LSURF/2)*2
      IF(KCOM.EQ.0)XNOS=-XNOS
      IF(LSURF-3)104,103,105
  103 I=1
      GOTO 110
  104 I=2
      GOTO 110
  105 IF(LSURF.LT.5)GOTO 103
      I=3
  110 CALL UN2(LDATA,IEMP(4),IEMP(1))
      LDATA=LDATA+1
```

```
      CALL UN2(LDATA,IEMP(2),IEMP(3))
      DO 115 J=1,3
      LH=IEMP(1)
      LV=IEMP(4)
      M=J-1
      IJK=LH+M
      IJK1=LV+M
      TEMP(J)=ASTER(IJK)+ASTER(IJK1)
      MK=J-1+IEMP(4)
      TEMP1(J)=ASTER(MK)
  115 CONTINUE
      CALL DCOSP(TEMP1,TEMP,WB)
      DO 120 J=1,3
      WB(J)=XNOS*WB(J)
  120 CONTINUE
      GOTO 1000
C
C     SPH
C
  150 CALL UN2(LDATA,LV,DUM)
      DO 160 I=1,3
      M=I-1+LV
      TEM(I)=ASTER(M)
  160 CONTINUE
      CALL DCOSP(XI,TEM,WB)
      DO 170 I=1,3
      WB(I)=XNOS*WB(I)
  170 CONTINUE
      GOTO 1000
C
C     RCC
C
  200 IF(LSURF-2)202,201,210
  201 XNOS=-XNOS
  202 CALL UN2(LDATA,LV1,LV2)
      DO 203 I=1,3
      M=I-1
      IJK1=M+LV1
      IJK2=M+LV2
      TEM(I)=ASTER(IJK1)
      TEM1(I)=ASTER(IJK1)+ASTER(IJK2)
  203 CONTINUE
      CALL DCOSP(TEM,TEM1,WB)
      DO 204 I=1,3
      WB(I)=XNOS*WB(I)
  204 CONTINUE
      GOTO 1000
C
C         DIR COS FOR NORMAL TO SURFACE ONE OR TWO
C         NOW HAVE TO GET FROM A POINT TO THE HEIGHT VECTOR
C
  210 CALL UN2(LDATA,LV,LH)
      LR1=MASTER(LDATA+1)
      DO 211 J=1,3
      M=J-1
      IJK=LV+M
      TEM(J)=ASTER(IJK)
      IJK1=LH+M
      TEM1(J)=ASTER(IJK)+ASTER(IJK1)
  211 CONTINUE
```

```
      CALL DCOSP(TEM,XI,WN)
      CALL DCOSP(TEM,TEM1,WI)
      SUM=0.
      DO 212 J=1,3
      SUM=SUM+WN(J)*WI(J)
  212 CONTINUE
      DO 214 J=1,3
      XP(J)=SUM*XDIST(TEM,XI)
      XP(J)=XP(J)*WI(J)+TEM(J)
  214 CONTINUE
      IF(ITYPE.EQ.5)GOTO 250
      CALL DCOSP(XI,XP,WB)
      DO 220 J=1,3
      WB(I)=XNOS*WB(J)
  220 CONTINUE
      GOTO 1000
C
C     REC
C
C     FOR SURFACE 1 AND 2 NORMAL IS SAME AS RCC
C     FOR SURFACE 3 JUMP OUT WHEN XP(I)=POINT ON HEIGHT VECTOR
C
  250 LDATA=LDATA+1
      CALL UN2(LDATA,LR1,LR2)
      DO 255 J=1,3
      M=J-1
      IJK1=M+LR1
      TEMP(J)=ASTER(IJK1)+XP(J)
      IJK2=M+LR2
      TEMP1(J)=ASTER(IJK2)+XP(J)
  255 CONTINUE
      A1=XDIST(XP,TEMP)
      A2=XDIST(XP,TEMP1)
      IF(A1.GE.A2)GOTO 260
      A1=A2
      A3=A1
      A2=A3
      TEMP(1)=TEMP1(1)
      TEMP(2)=TEMP1(2)
      TEMP(3)=TEMP1(3)
  260 C=SQRT(A1*A1-A2*A2)
      CALL DCOSP(XP,TEMP,WN)
      DO 265 J=1,3
      TEM(J)=XP(J)+C*WN(J)
      TEM1(J)=XP(J)-C*WN(J)
  265 CONTINUE
      CALL DCOSP(TEM,XI,WN)
      DO 270 J=1,3
      TEM(J)=2.*A1*WN(J)+TEM(J)
  270 CONTINUE
      CALL DCOSP(TEM,TEM1,WB)
      DO 275 J=1,3
      WB(J)=XNOS*WB(J)
  275 CONTINUE
      GOTO 1000
C
C     TRC
C
  300 IF(LSURF.LE.2)GO TO 320
      CALL UN2(LDATA,LV,LH)
```

```
      LDATA=LDATA+1
      CALL UN2(LDATA,LR1,LR2)
      DIF=ASTER(LR1)-ASTER(LR2)
      IF(DIF)302,301,303
  301 WRITE (6,905)
      STOP
  302 TEMP(1)=LR1
      LR1=LR2
      LR2=TEMP(1)
      DIF=ABS(DIF)
  303 FACTR=ASTER(LR1)/DIF
      DO 304 J=1,3
      M=J-1
      IJK=M+LV
      IJK1=M+LH
      TEMP1(J)=ASTER(IJK)
      TEMP(J)=ASTER(IJK)+FACTR*ASTER(IJK1)
  304 CONTINUE
      TDIS=XDIST(XI,TEMP)
      QDIS=XDIST(TEMP1,TEMP)
      CALL DCOSP(TEMP,XI,WN)
      CALL DCOSP(TEMP,TEMP1,WA)
      SUM=0.
      DO 310 J=1,3
      SUM=WN(J)*WA(J)+SUM
  310 CONTINUE
      QSUM=TDIS/SUM
      QPLS=QDIS-QSUM
      DO 311 J=1,3
      TEMP(J)=-QPLS*WA(J)+TEMP1(J)
  311 CONTINUE
      CALL DCOSP(XI,TEMP,WB)
      DO 312 J=1,3
      WB(J)=XNOS*WB(J)
  312 CONTINUE
      GOTO 1000
C
  320 IF(LSURF.EQ.2)XNOS=-XNOS
      CALL UN2(LDATA,LV,LH)
      DO 321 J=1,3
      M=J-1
      IJK=M+LV
      TEMP(J)=ASTER(IJK)
      IJK1=M+LH
      TEMP1(J)=ASTER(IJK)+ASTER(IJK1)
  321 CONTINUE
      CALL DCOSP(TEMP,TEMP1,WB)
      DO 322 J=1,3
      WB(J)=XNOS*WB(J)
  322 CONTINUE
      GOTO 1000
C
C     ELL
C
  350 CALL UN2(LDATA,LR1,LR2)
      LS=MASTER(LDATA+1)
      DO 352 J=1,3
      M=J-1
      IJK1=M+LR1
      IJK2=M+LR2
```

```
      TEM(J)=ASTER(IJK1)
      TEM1(J)=ASTER(IJK2)
  352 CONTINUE
      A=ASTER(LS)
      CALL DCOSP(TEM,XI,WN)
      DO 353 J=1,3
      TEM(J)=A*WN(J)+TEM(J)
  353 CONTINUE
      CALL DCOSP(TEM,TEM1,WB)
      DO 354 J=1,3
      WB(J)=XNOS*WB(J)
  354 CONTINUE
      GOTO 1000
C
C     RAW
C
C     THIS WILL SHAPE THE BOX FOR LSURF=1,3,5,6
C     JUMPS TO 100 TO INDICATE BOX PORTION
C
  400 IF(LSURF.EQ.2)GOTO 401
      IF(LSURF.NE.4)GOTO 100
      WRITE (6,906)
      STOP
  401 CALL UN2(LDATA,LV,LV1)
      LDATA=LDATA+1
      CALL UN2(LDATA,LV2,LV3)
      DO 410 J=1,3
      M=J-1
      IJK1=M+LV1
      IJK2=M+LV2
      TEMP(J)=ASTER(IJK1)
      XMID(J)=ASTER(IJK1)-ASTER(IJK2)
      IJK3=M+LV3
      TEM(J)=ASTER(IJK3)
  410 CONTINUE
      I=1
      J=2
      K=3
      LK=0
      DO 411 KK=1,3
      TEM1(I)=XMID(J)*TEM(K)-XMID(K)*TEM(J)
      LK=I
      I=J
      J=K
      K=LK
  411 CONTINUE
      SUM=0.
      DO 412 J=1,3
      SUM=TEM1(J)*TEMP(J)+SUM
  412 CONTINUE
      SUM=-SUM/ABS(SUM)
      TLK=TEM1(1)**2+TEM1(2)**2+TEM1(3)**2
      TLK=SQRT(TLK)
      DO 420 J=1,3
      WB(J)=XNOS*SUM*TEM1(J)/TLK
  420 CONTINUE
      GOTO 1000
C
C     ARB
C
```

```
[illegible]
[illegible]
[illegible]
```

```
      T3=DOT(TEMP,AUN)/TAU
      T4=DOT(TEMP,C)
      GAMMA=DDN/HDN
      EM=GAMMA*R4+(1.0-GAMMA)*R2
      T5=HDA/HDN
      T6=HDK/HDN
      DO 510 I=1,3
      WB(I)=XNOS*(T3*(AUN(I)-T5*NF(I))+
     1     T4*(Q(I)-T6*NF(I))-EM*(R4-R2)*NF(I)/HDN)
  510 CONTINUE
      CALL UNIT(WB)
      GOTO 1000
C
  520 IF(LSURF.EQ.2)XNOS=-XNOS
      CALL UN2(LDATA,LV,LH)
      LDATA=LDATA+1
      CALL UN2(LDATA,LN,DUM)
      DO 521 I=1,3
      J=LN+I-1
      WB(I)=XNOS*ASTER(J)
  521 CONTINUE
      GOTO 1000
C
C     TOR
C
  550 CALL UN2(LDATA,LV,LN)
      LDATA=LDATA+1
      CALL UN2(LDATA,LR1,DUM)
      DO 551 I=1,3
      J=I-1
      IJK=LV+J
      TEMP(I)=XI(I)-ASTER(IJK)
      IJK=LN+J
      TEMP1(I)=ASTER(IJK)
  551 CONTINUE
      R1=ASTER(LR1)
      CALL CROSS(TEM,TEMP1,TEMP)
      CALL CROSS(TEM1,TEM,TEMP1)
      CALL UNIT(TEM1)
      DO 552 I=1,3
      J=I-1
      IJK=LV+J
      TEM(I)=ASTER(IJK)
      TEMP1(I)=TEM(I)+R1*TEM1(I)
  552 CONTINUE
      CALL DCOSP(TEMP1,XI,WB)
      DO 553 I=1,3
      WB(I)=XNOS*WB(I)
  553 CONTINUE
      GOTO 1000
C
C     ARS
C
  600 NE=4
      IWH=MASTER(LDATA)
      INOW=IWH+8
      IEND=IWH+8+20*NE
      DO 610 I=1,3
      IJK=IWH+I+4
      ARSTP(I)=ASTER(IJK)
```

```
  610 CONTINUE                                                          CALC 489
      DTRAV=XDIST(ARSTP,XI)                                             CALC 490
  620 IF(ABS(DTRAV-ASTER(INOW)).GT.1.0E-07)GOTO 640                     CALC 491
      DO 630 I=1,3                                                      CALC 492
      IJK=INOW+I                                                        CALC 493
      WB(I)=ASTER(IJK)                                                  CALC 494
  630 CONTINUE                                                          CALC 495
      CALL UNIT(WB)                                                     CALC 496
      GOTO 1000                                                         CALC 497
C                                                                       CALC 498
  640 INOW=INOW+NE                                                      CALC 499
      IF(IEND.GT.INOW)GOTO 620                                          CALC 500
      WRITE (6,903)                                                     CALC 501
      STOP                                                              CALC 502
C                                                                       CALC 503
C     COMPUTE OBLIQUITY ANGLE                                           CALC 504
C     COMPUT NORMAL DIST  (SN)                                          CALC 505
C                                                                       CALC 506
 1000 DO 1001 J=1,3                                                     CALC 507
      XB(J)=XI(J)+WS(J)*1.0E-3                                          CALC 508
 1001 CONTINUE                                                          CALC 509
      ANGLE=0.                                                          CALC 510
      DO 1002 J=1,3                                                     CALC 511
      ANGLE=ANGLE+WB(J)*WS(J)                                           CALC 512
 1002 CONTINUE                                                          CALC 513
      IF(ABS(ANGLE).LE.1.)GOTO 1010                                     CALC 514
      ANGLE=0.                                                          CALC 515
      SN=0.                                                             CALC 516
      WRITE (6,904)NIR,ITYPE,NBO,LSURF,WB,WS,XP,XB,XI,XNOS              CALC 517
      IR=NIR                                                            CALC 518
      GOTO 40                                                           CALC 519
C                                                                       CALC 520
 1010 ANGLE=ATAN2(SQRT(1.-ANGLE*ANGLE),ANGLE)*180./3.141592654          CALC 521
      IF(ANGLE.LE.90.)GOTO 1020                                         CALC 522
      DO 1011 J=1,3                                                     CALC 523
      WB(J)=-WB(J)                                                      CALC 524
 1011 CONTINUE                                                          CALC 525
      GOTO 1000                                                         CALC 526
C                                                                       CALC 527
 1020 NASC=-2                                                           CALC 528
      IR=NIR                                                            CALC 529
      CALL GI(SI,IRPRIM,XP)                                             CALC 530
      SN=SI                                                             CALC 531
      GOTO 40                                                           CALC 532
      END                                                               CALC 533
C                                                                       CALC 534
C                                                                       CALC 535
C                                                                       CALC 536
C                                                                       CALC 537
C                                                                       CALC 538
      SUBROUTINE TESTG                                                  ****  24
C                                                                       TESTG  2
C     TESTG OPTIONS                                                     TESTG  3
C                                                                       TESTG  4
C         NRAYS 0 0     TRACE A RAY BETWEEN TWO GIVEN POINTS            TESTG  5
C                             XBS TO XBF                                TESTG  6
C                                                                       TESTG  7
      DIMENSION XP(3),XBF(3)                                            TESTG  8
      COMMON/PAREM/XB(3),WB(3),IR                                       TESTG  9
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 TESTG 10
```

```
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,      TESTG 11
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP                TESTG 12
      COMMON/WALT/LIRFO,NG1ERR                                            TESTG 13
C                                                                         TESTG 14
  901 FORMAT(2I10)                                                        TESTG 15
  902 FORMAT(1H0,22HNUMBER OF SPECIAL RAYS,I5)                            TESTG 16
  903 FORMAT(3E15.7,3I15)                                                 TESTG 17
  904 FORMAT(1H0,5HSTART,5X,4H XB=,3E15.7,8H IRSTRT=,I5/                  TESTG 18
     1    4H END,7X,4HXBF=,3E15.7,8H  IRFIN=,I5)                          TESTG 19
  905 FORMAT(1H0,3HWB=,3E15.7,5X,6HRANGE=,E15.7)                          TESTG 20
  906 FORMAT(1H0,8X,2HIR,4X,6HIRPRIM,12X,2HS1,13X,2HXP,13X,2HYP,          TESTG 21
     1    13X,2HZP,12X,4HDIST)                                            TESTG 22
  907 FORMAT(2I10,5X,5E15.7)                                              TESTG 23
  908 FORMAT(1H0,21HTROUBLE IN REGION IR=,I10)                            TESTG 24
C                                                                         TESTG 25
      READ (5,901)NRAYS,NG1ERR                                            TESTG 26
      WRITE (6,902)NRAYS                                                  TESTG 27
      IF(NG1ERR.LE.0)NG1ERR=25                                            TESTG 28
C                                                                         TESTG 29
      DO 50 IRAY=1,NRAYS                                                  TESTG 30
      READ (5,903)XB,IRSTRT                                               TESTG 31
      READ (5,903)XBF,IRFIN                                               TESTG 32
      WRITE (6,904)XB,IRSTRT,XBF,IRFIN                                    TESTG 33
      RANGE=XDIST(XB,XBF)                                                 TESTG 34
      CALL DCOSP(XB,XBF,WB)                                               TESTG 35
      WRITE (6,905)WB,RANGE                                               TESTG 36
      IR=IRSTRT                                                           TESTG 37
      NASC=-1                                                             TESTG 38
      WRITE (6,906)                                                       TESTG 39
C                                                                         TESTG 40
   10 CALL G1(S1,IRPRIM,XP)                                               TESTG 41
      IF(IERR.GE.NG1ERR)GOTO 60                                           TESTG 42
      WRITE (6,907)IR,IRPRIM,S1,XP,DIST                                   TESTG 43
      IF(DIST.GE.RANGE)GOTO 30                                            TESTG 44
      IF(IRPRIM.LE.0)GOTO 20                                              TESTG 45
      IR=IRPRIM                                                           TESTG 46
      GOTO 10                                                             TESTG 47
C                                                                         TESTG 48
   20 WRITE (6,908)IR                                                     TESTG 49
      GOTO 50                                                             TESTG 50
   30 IF(IR.NE.IRFIN)GOTO 20                                              TESTG 51
   50 CONTINUE                                                            TESTG 52
   60 IERR=0                                                              TESTG 53
      RETURN                                                              TESTG 54
      END.                                                                TESTG 55
C                                                                         TESTG 56
C                                                                         TESTG 57
C                                                                         TESTG 58
C                                                                         TESTG 59
      SUBROUTINE VOLUM                                                    ****  25
      DIMENSION VASTER(1000),WAB(3),WTB(3),WOB(3),DSP(3),                 VOLUM  2
     1  XV(3),XT(3),XA(3),XO(3),XP(3),XTEMP(3)                            VOLUM  3
      COMMON ASTER(30000)                                                 VOLUM  4
      COMMON/PAREM/XB(3),WB(3),IR                                         VOLUM  5
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                   VOLUM  6
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,       VOLUM  7
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP                VOLUM  8
      COMMON/WALT/LIRFO,NG1ERR                                            VOLUM  9
C                                                                         VOLUM 10
  901 FORMAT(3E20.8)                                                      VOLUM 11
```

```
  902 FORMAT(2E20.8)                                                    VOLUM 12
  903 FORMAT(1H0,10X,6HVERTEX,14X,6HTOP.PT,14X,6HBOT.PT,14X,7HSIDE.PT)  VOLUM 13
  904 FORMAT(4E20.8)                                                    VOLUM 14
  905 FORMAT(1H0, 8X,12HDELTA ON TOP,E20.8,10X,10HSIDE DELTA,E20.8)     VOLUM 15
  906 FORMAT(2I10)                                                      VOLUM 16
  908 FORMAT(1H0, 2X,18HSTARTING REGION IS,I5)                          VOLUM 17
  909 FORMAT(1H0,16HVASTER OVERWRITE,5X,6HNRMAX=,I5)                    VOLUM 18
  910 FORMAT(I10,E20.8)                                                 VOLUM 19
  911 FORMAT(1H0,8HBAD CARD/I10,E20.8,14H NOT PROCESSED)                VOLUM 20
  912 FORMAT(I10,E20.8,5X,E20.8,5X,E9.2)                                VOLUM 21
  913 FORMAT(1H0,5HSUMV=,5X,E20.8)                                      VOLUM 22
C                                                                       VOLUM 23
      READ (5,906)IR,NG1ERR                                             VOLUM 24
      IF(NG1ERR.LE.0)NG1ERR=25                                          VOLUM 25
      READ (5,901)(XV(I),I=1,3)                                         VOLUM 26
      READ (5,901)(XT(I),I=1,3)                                         VOLUM 27
      READ (5,901)(XO(I),I=1,3)                                         VOLUM 28
      READ (5,901)(XA(I),I=1,3)                                         VOLUM 29
      READ (5,902)DOD,DT                                                VOLUM 30
      WRITE (6,903)                                                     VOLUM 31
      WRITE (6,904)(XV(J),XT(J),XO(J),XA(J),J=1,3)                      VOLUM 32
      WRITE (6,905)DOD,DT                                               VOLUM 33
      WRITE (6,908)IR                                                   VOLUM 34
      IF(NRMAX.GT.2000)WRITE (6,909)NRMAX                               VOLUM 35
      CALL DCOSP(XV,XT,WTB)                                             VOLUM 36
      CALL DCOSP(XV,XO,WOB)                                             VOLUM 37
      CALL DCOSP(XV,XA,WAB)                                             VOLUM 38
      XVDIS=XDIST(XV,XA)                                                VOLUM 39
      TESTON=0.                                                         VOLUM 40
      TESTOV=0.                                                         VOLUM 41
      XTEMP(1)=0.                                                       VOLUM 42
      DO 10 I=1,NRMAX                                                   VOLUM 43
      VASTER(I)=0.                                                      VOLUM 44
   10 CONTINUE                                                          VOLUM 45
      JIR=IR                                                            VOLUM 46
      IRJ=IR                                                            VOLUM 47
      N2=XDIST(XV,XO)/DOD+1.                                            VOLUM 48
      N1=XDIST(XV,XT)/DT+1.                                             VOLUM 49
C                                                                       VOLUM 50
      DO 300 J=1,N2                                                     VOLUM 51
      DO 100 I=1,3                                                      VOLUM 52
      DSP(I)=WTB(I)*DT                                                  VOLUM 53
      XB(I)=XV(I)                                                       VOLUM 54
      WB(I)=WAB(I)                                                      VOLUM 55
  100 CONTINUE                                                          VOLUM 56
      S1=0.                                                             VOLUM 57
      IR=JIR                                                            VOLUM 58
      DO 200 I=1,N1                                                     VOLUM 59
      NASC=-1                                                           VOLUM 60
  110 CALL G1(S1,IRPRIM,XP)                                             VOLUM 61
      IF(IERR.GE.NG1ERR)GOTO 400                                        VOLUM 62
      VASTER(IR)=VASTER(IR)+S1                                          VOLUM 63
      IF(DIST.GE.XVDIS)GOTO 115                                         VOLUM 64
      IF(IRPRIM.LE.0)GOTO 120                                           VOLUM 65
      IR=IRPRIM                                                         VOLUM 66
      GOTO 110                                                          VOLUM 67
  115 VASTER(IR)=VASTER(IR)-(DIST-XVDIS)                                VOLUM 68
  120 XTEMP(1)=WB(1)                                                    VOLUM 69
      XTEMP(2)=WB(2)                                                    VOLUM 70
      XTEMP(3)=WB(3)                                                    VOLUM 71
```

```
      IR=JIR                                                            VOLUM 72
      TESTDN=TESTDN-DT                                                  VOLUM 73
      IF(TESTDN.GT.0.)GOTO 180                                          VOLUM 74
      WB(1)=WTB(1)                                                      VOLUM 75
      WB(2)=WTB(2)                                                      VOLUM 76
      WB(3)=WTB(3)                                                      VOLUM 77
      NASC=-1                                                           VOLUM 78
      CALL G1(S1,IRPRIM,XP)                                             VOLUM 79
      IF(IERR.GE.NG1ERR)GOTO 400                                        VOLUM 80
      IF(S1-DT)130,160,170                                              VOLUM 81
  130 IR=IRPRIM                                                         VOLUM 82
      JIR=IR                                                            VOLUM 83
      CALL G1(S1,IRPRIM,XP)                                             VOLUM 84
      IF(IERR.GE.NG1ERR)GOTO 400                                        VOLUM 85
      IF(DIST-DT)140,160,170                                            VOLUM 86
  140 IF(IRPRIM)150,210,130                                             VOLUM 87
  150 STOP                                                              VOLUM 88
  160 IR=IRPRIM                                                         VOLUM 89
      JIR=IR                                                            VOLUM 90
  170 TESTDN=S1                                                         VOLUM 91
  180 DO 190 JI=1,3                                                     VOLUM 92
      WB(JI)=XTEMP(JI)                                                  VOLUM 93
      XB(JI)=XB(JI)+DSP(JI)                                             VOLUM 94
  190 CONTINUE                                                          VOLUM 95
  200 CONTINUE                                                          VOLUM 96
C                                                                       VOLUM 97
C     ONE PLANE DONE - MOVE IN FOR NEXT PLANE IN LINE                   VOLUM 98
C                                                                       VOLUM 99
  210 NASC=-1                                                           VOLUM100
      DO 220 I=1,3                                                      VOLUM101
      WB(I)=WOB(I)                                                      VOLUM102
      XB(I)=XV(I)                                                       VOLUM103
  220 CONTINUE                                                          VOLUM104
      JIR=IRJ                                                           VOLUM105
      IR=JIR                                                            VOLUM106
      TESTDN=0.                                                         VOLUM107
      TESTOV=TESTOV-DOD                                                 VOLUM108
      IF(TESTOV)230,230,280                                             VOLUM109
  230 CALL G1(S1,IRPRIM,XP)                                             VOLUM110
      IF(IERR.GE.NG1ERR)GOTO 400                                        VOLUM111
      IF(S1-DOD)240,260,270                                             VOLUM112
  240 IR=IRPRIM                                                         VOLUM113
      IRJ=IR                                                            VOLUM114
      CALL G1(S1,IRPRIM,XP)                                             VOLUM115
      IF(IERR.GE.NG1ERR)GOTO 400                                        VOLUM116
      IF(DIST-DOD)250,260,270                                           VOLUM117
  250 IF(IRPRIM)150,400,230                                             VOLUM118
  260 IR=IRPRIM                                                         VOLUM119
      IRJ=IR                                                            VOLUM120
  270 TESTOV=S1                                                         VOLUM121
  280 DO 290 I=1,3                                                      VOLUM122
      XA(I)=XA(I)+WOB(I)*DOD                                            VOLUM123
      XV(I)=XV(I)+WOB(I)*DOD                                            VOLUM124
      XT(I)=XT(I)+WOB(I)*DOD                                            VOLUM125
  290 CONTINUE                                                          VOLUM126
      JIR=IR                                                            VOLUM127
  300 CONTINUE                                                          VOLUM128
C                                                                       VOLUM129
C     VOLUMES COMPUTED                                                  VOLUM130
C                                                                       VOLUM131
```

```
  400 READ (5,910)IR1,VR                                                VOLUM132
      IF(IERR.GE.NG1ERR)GOTO 500                                        VOLUM133
      IF(IR1.LE.0)IR1=NRMAX+1                                           VOLUM134
      SUMV=0.                                                           VOLUM135
C                                                                       VOLUM136
      DO 450 I=1,NRMAX                                                  VOLUM137
      VASTER(I)=VASTER(I)*DOD*DT                                        VOLUM138
      IF(I-IR1)410,430,420                                              VOLUM139
  410 WRITE (6,910)I,VASTER(I)                                          VOLUM140
      GOTO 440                                                          VOLUM141
  420 WRITE (6,911)IR1,VR                                               VOLUM142
      READ (5,910)IR1,VR                                                VOLUM143
      GOTO 410                                                          VOLUM144
C          VOLUME REPLACEMENT                                           VOLUM145
  430 XPERC=100.*(VASTER(I)/VR-1.)                                      VOLUM146
      WRITE (6,912)I,VASTER(I),VR,XPERC                                 VOLUM147
      VASTER(I)=VR                                                      VOLUM148
      READ (5,910)IR1,VR                                                VOLUM149
  440 SUMV=SUMV+VASTER(I)                                               VOLUM150
  450 CONTINUE                                                          VOLUM151
      WRITE (6,913)SUMV                                                 VOLUM152
  500 IERR=0                                                            VOLUM153
      RETURN                                                            VOLUM154
      END                                                               VOLUM155
C                                                                       VOLUM156
C                                                                       VOLUM157
C                                                                       VOLUM158
C                                                                       VOLUM159
      SUBROUTINE AREA                                                   ****  26
      DIMENSION XP(3),WP(3),XBS(3),CONVRT(4,4),TYPEUN(4)                AREA   2
      COMMON ASTER(30000)                                               AREA   3
      COMMON/PARAM/XB(3),WB(3),IR                                       AREA   4
       COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                AREA   5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     AREA   6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              AREA   7
      COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),              AREA   8
     1  TRAVEL,SN,V,H,IVIH                                              AREA   9
      COMMON/WALT/LIRFO,NG1ERR                                          AREA  10
      COMMON/CELL/CELSIZ                                                AREA  11
      COMMON/ENGEOM/LLGEOM                                              AREA  12
C                                                                       AREA  13
  901 FORMAT(/I10,6X,2A2)                                               AREA  14
  902 FORMAT(6E12.8)                                                    AREA  15
  908 FORMAT(1H0,22HMEMORY OVERLAP IN AREA,5X,7HLEGEOM=,I6,             AREA  16
     1    5X,6HLAREA=,I6,5X,6HLIRFO=,I6)                                AREA  17
  909 FORMAT(1H0,13HERROR IN AREA,5X,9HICODE = 0)                       AREA  18
  910 FORMAT(1H0,8HAZIMUTH=,F10.3,5X,10HELEVATION=,F10.3)               AREA  19
  911 FORMAT(1H0,12HCELL SIZE IS,F4.1,1X,1HX,F4.1,1X,A2,1H.,10X,        AREA  20
     1    12HAREAS IN SQ.,1X,A2,1H.)                                    AREA  21
  912 FORMAT(1H0,5HICODE,19X,4HAREA/)                                   AREA  22
  913 FORMAT(I5,15X,F12.5)                                              AREA  23
  914 FORMAT(1H0,15HPRESENTED AREA=,F12.5)                              AREA  24
  915 FORMAT(1H0,18HNUMBER OF CELLS IS,I5,10X,                          AREA  25
     1    22HNUMBER OF CELLS HIT IS,I5)                                 AREA  26
C                                                                       AREA  27
C          IN=1    FT=2    CM=3    M=4                                  AREA  28
C              1 SQ. M. = 39.37 * 39.37 SQ. IN.                         AREA  29
C                                                                       AREA  30
      DATA HHIN,HHFT,HHCM,HHMB,HHBB/2HIN,2HFT,2HCM,2HM ,2H  /           AREA  31
      TYPEUN(1)=HHIN                                                    AREA  32
```

```
      TYPEUN(2)=HHFT                                                    AREA  33
      TYPEUN(3)=HHCM                                                    AREA  34
      TYPEUN(4)=HHMB                                                    AREA  35
      CONVRT(1,1)=1.                                                    AREA  36
      CONVRT(1,2)=.006944444444444444                                   AREA  37
      CONVRT(1,3)=6.451625806                                           AREA  38
      CONVRT(1,4)=.0006451625806                                        AREA  39
      CONVRT(2,1)=144.                                                  AREA  40
      CONVRT(2,2)=1.                                                    AREA  41
      CONVRT(2,3)=929.0341161                                           AREA  42
      CONVRT(2,4)=.09290341161                                          AREA  43
      CONVRT(3,1)=.15499969                                             AREA  44
      CONVRT(3,2)=.001076386736                                         AREA  45
      CONVRT(3,3)=1.                                                    AREA  46
      CONVRT(3,4)=.0001                                                 AREA  47
      CONVRT(4,1)=1549.9969                                             AREA  48
      CONVRT(4,2)=10.7636736                                            AREA  49
      CONVRT(4,3)=10000.                                                AREA  50
      CONVRT(4,4)=1.                                                    AREA  51
      BLANK=HHBB                                                        AREA  52
C                                                                       AREA  53
      LAREA=LIRFO-1000                                                  AREA  54
      IF(LAREA.GE.LEGEOM)GOTO 10                                        AREA  55
      WRITE (6,908)LEGEOM,LAREA,LIRFO                                   AREA  56
      STOP                                                              AREA  57
   10 LAREA1=LIRFO-1                                                    AREA  58
      DO 20 L=LAREA,LAREA1                                              AREA  59
      ASTER(L)=0.                                                       AREA  60
   20 CONTINUE                                                          AREA  61
C                                                                       AREA  62
      READ (5,901)NX,NY,IRSTRT ,IENC,NG1ERR,NSTART,NEND,CELLUN,AREAUN   AREA  63
      READ (5,902)A,E,ENGTH,ZSHIFT,GROUND                               AREA  64
      READ (5,902)XSHIFT,YSHIFT,CELSIZ                                  AREA  65
      IF(IRSTRT .LE.0)IRSTRT=1                                          AREA  66
      IF(CELSIZ .LE.0.)CELSIZ=4.                                        AREA  67
      IF(NSTART.LE.0)NSTART=1                                           AREA  68
      IF(NG1ERR.LE.0)NG1ERR=25                                          AREA  69
      IF(AREAUN.EQ.BLANK)AREAUN=HHIN                                    AREA  70
      IF(CELLUN.EQ.BLANK)CELLUN=HHIN                                    AREA  71
      DO 30 I=1,4                                                       AREA  72
      IF(CELLUN.EQ.TYPEUN(I))GOTO 40                                    AREA  73
   30 CONTINUE                                                          AREA  74
   40 DO 50 J=1,4                                                       AREA  75
      IF(AREAUN.EQ.TYPEUN(J))GOTO 60                                    AREA  76
   50 CONTINUE                                                          AREA  77
   60 AREA=CELSIZ *CELSIZ *CONVRT(I,J)                                  AREA  78
C                                                                       AREA  79
      RADIAN=.017453292519943                                           AREA  80
      AR=A*RADIAN                                                       AREA  81
      ER=E*RADIAN                                                       AREA  82
      SA=SIN(AR)                                                        AREA  83
      CA=COS(AR)                                                        AREA  84
      SE=SIN(ER)                                                        AREA  85
      CE=COS(ER)                                                        AREA  86
      KL=NX*NY                                                          AREA  87
      NHIT=0                                                            AREA  88
C                                                                       AREA  89
C      PROCESS KL CELLS IN GRID                                         AREA  90
C                                                                       AREA  91
      DO 200 KK=NSTART,KL                                               AREA  92
```

```
      WB(1)=-CE*CA                                                      AREA  93
      WB(2)=-CE*SA                                                      AREA  94
      WB(3)=-SE                                                         AREA  95
      II=((KK-1)/NX)+1                                                  AREA  96
      J=KK-(II-1)*NX                                                    AREA  97
C         COMPUTE COORDINATES OF GRID CELL IN GRID PLANE                AREA  98
      CELL2=.5*CELSIZ                                                   AREA  99
      V=FLOAT((NY/2)-II)*CELSIZ +CELL2                                  AREA 100
      VREF=V+CELL2                                                      AREA 101
      H=FLOAT((NX/2)- J)*CELSIZ +CELL2                                  AREA 102
      HREF=H+CELL2                                                      AREA 103
      IV=RAN(-1)*10.                                                    AREA 104
      IH=RAN(-1)*10.                                                    AREA 105
      IVIH=10*IH+IV                                                     AREA 106
C         COMPUTE H,V AT RANDOM POINT IN GRID CELL                      AREA 107
      V=V+CELSIZ *FLOAT(IV)/10.+CELSIZ /20.                             AREA 108
      H=H+CELSIZ *FLOAT(IH)/10.+CELSIZ /20.                             AREA 109
C         X,Y,Z IN COORDINATE SYSTEM OF VEHICLE                         AREA 110
      XBS(1)=XSHIFT-V*CA*SE-H*SA                                        AREA 111
      XBS(2)=YSHIFT-V*SA*SE+H*CA                                        AREA 112
      XBS(3)=ZSHIFT+V*CE                                                AREA 113
      CALL TRANPIC(WP)                                                  AREA 114
      XBS(1)=XBS(1)+WP(1)*1.0E-4                                        AREA 115
      XBS(2)=XBS(2)+WP(2)*1.0E-4                                        AREA 116
      XBS(3)=XBS(3)+WP(3)*1.0E-4                                        AREA 117
      XB(1)=XBS(1)-LNGTH*WB(1)                                          AREA 118
      XB(2)=XBS(2)-LNGTH*WB(2)                                          AREA 119
      XB(3)=XBS(3)-LNGTH*WB(3)                                          AREA 120
      IF(XB(3).LE.GROUND)GOTO 200                                       AREA 121
C                                                                       AREA 122
C     TRACE RAY TO FIRST COMPONENT HIT                                  AREA 123
C                                                                       AREA 124
      IR=IRSTRT                                                         AREA 125
      NASC=-1                                                           AREA 126
  110 CALL GI(SI,IRPRIM,XP)                                             AREA 127
      IF(IERR.GE.NGIERR)RETURN                                          AREA 128
      IF(IRPRIM.LT.0)GOTO 200                                           AREA 129
      IF(NASC.LE.NRPP)IRPRIM=0                                          AREA 130
      IF(IRPRIM.EQ.0)GOTO 200                                           AREA 131
      LOC=LIREO+IRPRIM-1                                                AREA 132
      CALL ON2(LOC,ICODE,IDENT)                                         AREA 133
      IDENT=IDENT-1                                                     AREA 134
      IF(IDENT-(IDENT/10)*10.EQ.0)GOTO 120                              AREA 135
      IR=IRPRIM                                                         AREA 136
      GOTO 110                                                          AREA 137
  120 IF(ICODE.NE.0)GOTO 130                                            AREA 138
      WRITE (6,909)                                                     AREA 139
      GOTO 200                                                          AREA 140
  130 LOC=LAREA+ICODE-1                                                 AREA 141
      ASTER(LOC)=ASTER(LOC)+AREA                                        AREA 142
      NHIT=NHIT+1                                                       AREA 143
  200 CONTINUE                                                          AREA 144
C                                                                       AREA 145
C     PRINT RESULTS                                                     AREA 146
C                                                                       AREA 147
      WRITE (6,910)A,E                                                  AREA 148
      WRITE (6,911)CELSIZ, CELSIZ, CELLUN,AREAUN                        AREA 149
      WRITE (6,912)                                                     AREA 150
      SUMA=0.                                                           AREA 151
      DO 250 I=1,999                                                    AREA 152
```

```
      LOC=LAREA+I-1                                                     AREA 153
      IF(ASTER(LOC).EQ.0.)GOTO 250                                      AREA 154
      WRITE (6,913)I,ASTER(LOC)                                         AREA 155
      SUMA=SUMA+ASTER(LOC)                                              AREA 156
  250 CONTINUE                                                          AREA 157
      WRITE (6,914)SUMA                                                 AREA 158
      WRITE (6,915)KL,NHIT                                              AREA 159
      RETURN                                                            AREA 160
      END                                                               AREA 161
C                                                                       AREA 162
C                                                                       AREA 163
C                                                                       AREA 164
C                                                                       AREA 165
      SUBROUTINE G1(S1,IRPRIM,XP)                                       **** 27
C                                                                       G1   2
C     MAIN RAY TRACKING ROUTINE                                         G1   3
C                                                                       G1   4
C     GIVEN A RAY IN REGION IR AT POINT XB WITH DIRECTION               G1   5
C         COSINES WB; FIND THE DISTANCE (S1) TO THE NEXT REGION         G1   6
C         AND THE NUMBER OF THAT REGION (IRPRIM)                        G1   7
C                                                                       G1   8
C     NASC=-2            $$  CALL FROM CALC TO FIND NORMAL DIST         G1   9
C     NASC=-1            $$  START NEW RAY                              G1  10
C     IVOLUM=1           $$  CALL FROM VOLUM                            G1  11
C     ITESTG=1           $$  CALL FROM TESTG                            G1  12
C     DIST               $$  TOTAL DIST TRAVELED BY RAY SO FAR          G1  13
C                                                                       G1  14
      DIMENSION MASTER(30000),XP(3),XBD(3),LSURT(50),NASCT(50)          G1  15
      COMMON ASTER(30000)                                               G1  16
      COMMON/PAREM/XB(3),WB(3),IR                                       G1  17
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 G1  18
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     G1  19
     1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP               G1  20
      COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),TRAVEL,       G1  21
     1 SN,V,H,IVIII                                                     G1  22
      COMMON/WALT/LIRFO,NGIERR                                          G1  23
      COMMON/LSU/LSURF                                                  G1  24
      COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES     G1  25
      COMMON/DAVIS/IGRID,LOOP,INORM                                     G1  26
      COMMON/CELL/CELSIZ                                                G1  27
      EQUIVALENCE (ASTER,MASTER)                                        G1  28
C                                                                       G1  29
  901 FORMAT(1H0,32HERROR IN G1 AT 140      BAD ITYPE,5X,4HITY=,I5)     G1  30
  902 FORMAT(1H0,33HERROR IN G1 AT 510      SM 0= PINF,5X,3HIR=,I5)     G1  31
  903 FORMAT(4H XB=,3E20.8/4H WB=,3E20.8/10X,5HKLOOP,12X,3HNBO,         G1  32
     1 12X,3HLRI,12X,3HLRO,11X,4HNHIT,11X,4HLOOP/6I15)                  G1  33
  904 FORMAT(1H1,15(2H* ),3X, 9HERROR NO.,I5,3X,15(2H *)//)             G1  34
  905 FORMAT(34X,4HCELL,2I4)                                            G1  35
  906 FORMAT(19H ERROR IN G1 AT 640//4H J1=,I10,4H J2=,I10,7H LSURF=,   G1  36
     1 I10,6H NASC=,I10,4H IR=,I10/4H SM=,E21.10,4H S1=,E17.10/         G1  37
     2 4H WB=,3E21.10/4H XB=,3E21.10)                                   G1  38
  907 FORMAT(50H THE (SOLID POSITION/DEPTH/POINT NOW AT) IS ONE OF,     G1  39
     1 6H THESE/6H XBD =,3E21.10/6H DIST=,E21.10//)                     G1  40
  908 FORMAT(9X,3HRIN,12X,4HROUT,7X,8HENTERING,2X,7HLEAVING,3X,         G1  41
     1 8HBODY NO.,5X,3HRAY,/35X,8HSIDE NO.,2X,8HSIDE NO.//)             G1  42
  910 FORMAT(//16H TILT  RIN=ROUT=,E20.10,30X,2HI=,I5//)                G1  43
  911 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,7HSTARTED/)               G1  44
  912 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,7HHAS HIT/)               G1  45
  913 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,7HLEAVING/)               G1  46
  914 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,7H IN   /)                G1  47
```

```
915 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,8HENTERING/)
916 FORMAT(2(2X,E15.8),4X,I2,8X,I2,6X,I5,5X,8HWILL HIT/)
917 FORMAT(//4(14H END ERROR NO.,I4,3X)/)
918 FORMAT(1H0,I5,21H ERRORS IN G1, RETURN)

      INORM=0
      IF(NASC.EQ.-2)INORM=1
      S1=0.
      IF(NASC.GT.0)GOTO 20
C         NEW RAY
      DIST=0.
      IF(KLOOP.LT.32000)GOTO 15
      KLOOP=0
      LION=LIO+NBODY+NRPP-1
      DO 10 I=LIO,LION
      MASTER(I)=0
   10 CONTINUE
   15 KLOOP=KLOOP+1
C
C     BEGIN TRACING RAY
C
   20 SM=PINF
      NHIT=0
      LOC=LREGD+IR-1
      CALL UN2(LOC,LOC,NC)
      LOC=LOC-1
C
C     NC=NUM OF BODIES IN REGION DESCRIPTION
C     FIND RIN AND ROUT FOR EACH OF THESE BODIES
C         RIN  IS DIST FROM XB TO POINT WHERE RAY ENTERS THE BODY
C         ROUT IS DIST FROM XB TO POINT WHERE RAY LEAVES THE BODY
C              IF ROUT = -PINF RAY DOES NOT HIT BODY
C         G1 SELECTS SMALLEST OF RIN AND ROUT DISTANCES O DIST
C         1)  UNIQUE RIN VALUE - NEXT BODY IN PATH OF RAY
C         2)  2 OR MORE RIN VALUES - 2 OR MORE BODIES HAVE
C             A COMMON SURFACE
C         3)  ROUT FOR CURRENT BODY MEANS RAY WILL LEAVE
C             THIS BODY BEFORE ENCOUNTERING ANOTHER
C
      DO 500 N=1,NC
      LOC=LOC+1
      CALL UN2(LOC,DUM,NBO)
      ITEMP=LIO+NBO-1
      CALL UN3(ITEMP,LRI,LRO,LOOP)
      ITEMP=LBODY+3*(NBO-1)
      CALL UN2(ITEMP,ITYPE,LOCDA)
      IF(LOOP.NE.KLOOP)GOTO 130
C         CONTINUATION OF RAY
      IF(ITYPE.GT.11)GOTO 140
      IJK=LRIN+NBO-1
      RIN=ASTER(IJK)
      IJK=LROT+NBO-1
      ROUT=ASTER(IJK)
      IF(ITYPE.LT.10)GOTO 320
C         TOR AND ARS
C         IF DIST .GE. ROUT COMPUTE RIN / ROUT SET
      IF(ROUT.LT.0.)GOTO 320
      IF(DIST.LT.ROUT)GOTO 320
      IF(NASC.EQ.NBO)NASC=0
C
```

```
  130 LRI=1                                                          G1   108
      LRO=1                                                          G1   109
      ITY=ITYPE+1                                                    G1   110
      IF(ITY.GE.1.AND.ITY.LE.12)GOTO 200                             G1   111
  140 IERR=IERR+1                                                    G1   112
      WRITE (6,901)ITYPE                                             G1   113
      GOTO 800                                                       G1   114
C                                                                    G1   115
C          RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS           G1   116
  200 GOTO(205,210,215,220,225,230,235,240,245,250,255,260),ITY      G1   117
  205 CALL RPP(NBO)                                                  G1   118
      GOTO 300                                                       G1   119
  210 CALL BOX                                                       G1   120
      GOTO 300                                                       G1   121
  215 CALL SPH                                                       G1   122
      GOTO 300                                                       G1   123
  220 CALL RCC                                                       G1   124
      GOTO 300                                                       G1   125
  225 CALL REC                                                       G1   126
      GOTO 300                                                       G1   127
  230 CALL TRC                                                       G1   128
      GOTO 300                                                       G1   129
  235 CALL ELL                                                       G1   130
      GOTO 300                                                       G1   131
  240 CALL RAW                                                       G1   132
      GOTO 300                                                       G1   133
  245 CALL ARB                                                       G1   134
      GOTO 300                                                       G1   135
  250 CALL TEC                                                       G1   136
      GOTO 300                                                       G1   137
  255 CALL TOR                                                       G1   138
      GOTO 300                                                       G1   139
  260 CALL ARS                                                       G1   140
C                                                                    G1   141
  300 IJK=LRIN+NBO-1                                                 G1   142
      ASTER(IJK)=RIN                                                 G1   143
      IJK=LROT+NBO-1                                                 G1   144
      ASTER(IJK)=ROUT                                                G1   145
      IJK=LIO+NBO-1                                                  G1   146
      MASTER(IJK)=KLOOP+I15*(LRO+64*LRI)                             G1   147
C                                                                    G1   148
  320 IF(NASC.NE.NBO)GOTO 330                                        G1   149
      IF(LSURF)500,500,340                                           G1   150
C                                                                    G1   151
  330 IF(ROUT.LE.0.)GOTO 500                                         G1   152
      IF(RIN.GT.0.)GOTO 350                                          G1   153
C                                                                    G1   154
  340 IF(ABS(ROUT-SM).GT.SM*1.0E-6)GOTO 341                          G1   155
      ROUT=SM                                                        G1   156
      IJK=LROT+NBO-1                                                 G1   157
      ASTER(IJK)=ROUT                                                G1   158
      GOTO 345                                                       G1   159
  341 IF(ROUT-SM)342,345,500                                         G1   160
  342 IF(DIST.GE.ROUT)GOTO 500                                       G1   161
      NHIT=0                                                         G1   162
  345 NHIT=NHIT+1                                                    G1   163
      SM=ROUT                                                        G1   164
      LSURT(NHIT)=-LRO                                               G1   165
      NASCI(NHIT)=NBO                                                G1   166
      GOTO 500                                                       G1   167
```

```
C
  350 IF(ABS(RIN-SM).GT.SM*1.0E-6)GOTO 351
      RIN=SM
      IJK=LRIN+NBO-1
      ASTER(IJK)=RIN
      GOTO 355
  351 IF(RIN-SM)352,355,500
  352 IF(DIST.GE.RIN)GOTO 340
      NHIT=0
  355 NHIT=NHIT+1
      SM=RIN
      LSURT(NHIT)=LRI
      NASCT(NHIT)=NBO
C
  500 CONTINUE
C
C     SM.GE.PINF   ERROR AT 510 IN G1
C
      IF(SM.LT.PINF)GOTO 530
      WRITE (6,902)IR
      WRITE (6,903)XB,WB,KLOOP,NBO,LRI,LRO,NHIT,LOOP
      GOTO 700
C
  530 S1=S1+SM-DIST
      DIST=SM
      XP(1)=XB(1)+SM*WB(1)
      XP(2)=XB(2)+SM*WB(2)
      XP(3)=XB(3)+SM*WB(3)
C
      IF(NASC.EQ.-2)RETURN
C
C     FIND NEXT REGION (IRPRIM)
C
      DO 640 NN=1,NHIT
      NASC =NASCT(NN)
      LSURF=LSURT(NN)
      LTRUE=0
      LOC=LBODY+3*(NASC-1)
      LOC=LOC+1
      CALL UN2(LOC,LENT,LEAV)
      LOC=LOC+1
      CALL UN2(LOC,NENT,NEAV)
      IF(LSURF.LE.0)GOTO 600
      J1=LENT
      J2=LENT+NENT-1
      GOTO 610
  600 J1=LEAV
      J2=LEAV+NEAV-1
C
  610 IRPRIM=MASTER(J2)
      IF(J1.LE.J2)GOTO 620
      IF(NASC.GT.ARPP)GOTO 700
      IF(LSURF)630,700,700
C
  620 DO 625 J=J1,J2
      IRPRIM=MASTER(J)
      CALL WOWI(IRPRIM,LSURF,NASC,LTRUE)
      IF(LTRUE.GT.0)GOTO 650
  625 CONTINUE
C
```

```
C     RPP CHECK                                                         G1   228
C                                                                       G1   229
      IF(NASC.GT.NRPP)GOTO 640                                          G1   230
      IF(LSURF)630,700,640                                              G1   231
  630 CALL RPP2(LSURF,XP,IRP)                                           G1   232
      IF(IRP.GT.0)GOTO 631                                              G1   233
      IRPRIM=0                                                          G1   234
      RETURN                                                            G1   235
C                                                                       G1   236
  631 LTRUE=0                                                           G1   237
      LOC=LBODY+3*(IRP-1)                                               G1   238
      LOC=LOC+1                                                         G1   239
      CALL UN2(LOC,LENT,LEAV)                                           G1   240
      LOC=LOC+1                                                         G1   241
      CALL UN2(LOC,NENT,NEAV)                                           G1   242
      J1=LENT                                                           G1   243
      J2=LENT+NENT-1                                                    G1   244
      IF(J1.GT.J2)GOTO 700                                              G1   245
C                                                                       G1   246
      DO 632 J=J1,J2                                                    G1   247
      IRPRIM=MASTER(J)                                                  G1   248
      CALL WOWI(IRPRIM,LSURF,IRP,LTRUE)                                 G1   249
      IF(LTRUE.GT.0)GOTO 650                                            G1   250
  632 CONTINUE                                                          G1   251
C                                                                       G1   252
  640 CONTINUE                                                          G1   253
      GOTO 700                                                          G1   254
C                                                                       G1   255
C     NEXT REGION (IRPRIM) HAS BEEN DETERMINED                          G1   256
C                                                                       G1   257
  650 IF(IR.EQ.IRPRIM)GOTO 660                                          G1   258
      IF(S1.EQ.0.)GOTO 660                                              G1   259
      IF(S1.LT.0.)GOTO 700                                              G1   260
      IF(ABS(S1).LE.1.0E-6)GOTO 660                                     G1   261
      IF(IVOLUM.EQ.IYES)RETURN                                          G1   262
      IF(ITESTG.EQ.IYES)RETURN                                          G1   263
      LOC=LIRFO+IR-1                                                    G1   264
      CALL UN2(LOC,ICODE,IDENT)                                         G1   265
      LOC=LIRFO+IRPRIM-1                                                G1   266
      CALL UN2(LOC,ICODE1,IDENT1)                                       G1   267
      IF(IDENT.EQ.1)GOTO 655                                            G1   268
      IF(IDENT.EQ.IDENT1)GOTO 660                                       G1   269
      RETURN                                                            G1   270
  655 IF(ICODE.NE.ICODE1)RETURN                                         G1   271
  660 IR=IRPRIM                                                         G1   272
      GOTO 20                                                           G1   273
C                                                                       G1   274
C     DIAGNOSTIC ERROR PRINT                                            G1   275
C                                                                       G1   276
  700 IERR=IERR+1                                                       G1   277
      WRITE (6,904)IERR                                                 G1   278
      IF(IVOLUM.EQ.IYES.OR.ITESTG.EQ.IYES)GOTO 705                      G1   279
      IH=ABS(H/CELSIZ )+.5                                              G1   280
      IF(H.LT.0.)IH=-IH                                                 G1   281
      IV=ABS(V/CELSIZ )+.5                                              G1   282
      IF(V.LT.0.)IV=-IV                                                 G1   283
      WRITE (6,905)IH,IV                                                G1   284
  705 WRITE (6,906)J1,J2,LSURF,NASC,IR,SM,S1,WB,XB                      G1   285
      XBD(1)=XB(1)-DIST                                                 G1   286
      XBD(2)=XB(2)-DIST                                                 G1   287
```

```
      XBD(3)=XB(3)-DIST                                                 G1   288
      WRITE (6,907)XBD,DIST                                             G1   289
      WRITE (6,908)                                                     G1   290
      NN=NBODY+NRPP                                                     G1   291
C                                                                       G1   292
      DO 750 I=1,NN                                                     G1   293
      LOC=LIO+I-1                                                       G1   294
      CALL UN3(LOC,I1,I2,I3)                                            G1   295
      IF(KLOOP.NE.I3)GOTO 750                                           G1   296
      IJK=LRIN+I-1                                                      G1   297
      RIN=ASTER(IJK)                                                    G1   298
      IJK=LROT+I-1                                                      G1   299
      ROUT=ASTER(IJK)                                                   G1   300
      IF(RIN.NE.ROUT)GOTO 710                                           G1   301
      WRITE (6,910)RIN,I                                                G1   302
      GOTO 750                                                          G1   303
C                                                                       G1   304
  710 IF(ABS(RIN).NE.PINF)GOTO 720                                      G1   305
      IF(ABS(ROUT)-PINF)740,750,740                                     G1   306
  720 IF(RIN-DIST)730,744,745                                           G1   307
  730 IF(ROUT-DIST)741,742,743                                          G1   308
C                                                                       G1   309
  740 WRITE (6,911)RIN,ROUT,I1,I2,I                                     G1   310
      GOTO 750                                                          G1   311
  741 WRITE (6,912)RIN,ROUT,I1,I2,I                                     G1   312
      GOTO 750                                                          G1   313
  742 WRITE (6,913)RIN,ROUT,I1,I2,I                                     G1   314
      GOTO 750                                                          G1   315
  743 WRITE (6,914)RIN,ROUT,I1,I2,I                                     G1   316
      GOTO 750                                                          G1   317
  744 WRITE (6,915)RIN,ROUT,I1,I2,I                                     G1   318
      GOTO 750                                                          G1   319
  745 WRITE (6,916)RIN,ROUT,I1,I2,I                                     G1   320
C                                                                       G1   321
  750 CONTINUE                                                          G1   322
      WRITE (6,917)IERR,IERR,IERR,IERR                                  G1   323
      IRPRIM=-1                                                         G1   324
C                                                                       G1   325
  800 IF(IERR.GE.NG1ERR)WRITE (6,918)NG1ERR                             G1   326
      RETURN                                                            G1   327
      END                                                               G1   328
C                                                                       G1   329
C                                                                       G1   330
      SUBROUTINE WOWI(JREG,LSURF,NEX,LTRUE)                             ****  28
C                                                                       WOWI   2
C     GIVEN A POINT (XB) AND A REGION (JREG), DOES XB                   WOWI   3
C         LIE WITHIN JREG                                               WOWI   4
C                                                                       WOWI   5
C     SUFFICIENT CONDITION FOR POINT XB TO BE IN REGION                 WOWI   6
C         JREG, IS THAT REGION DESCRIPTION OF JREG BE                   WOWI   7
C         SATISFIED. TWO REGIONS CANNOT BE SATISFIED FOR                WOWI   8
C         THE SAME POINT                                                WOWI   9
C                                                                       WOWI  10
C     + OPERATOR VALID IF ROUT.GT.0 AND RIN.LE.DIST.LT.ROUT             WOWI  11
C     - OPERATOR VALID IF ROUT.LE.0 OR DIST.LT.RIN OR DIST.GE.ROUT      WOWI  12
C     OR OPERATOR VALID IF ALL (+) AND (-) IN (OR) STATEMENT VALID      WOWI  13
C                                                                       WOWI  14
C     REGION DESCRIPTION WITH 1 OR MORE (OR) STATEMENTS VALID           WOWI  15
C         IF ANY ONE OF (OR) STATEMENTS IS VALID                        WOWI  16
C     REGION DESCRIPTION WITH NO (OR) STATEMENTS IS VALID ONLY          WOWI  17
```

```
C          IF EVERY (+) AND (-) OPERATOR IS VALID                       WOWI  18
C                                                                       WOWI  19
      DIMENSION MASTER(30000)                                           WOWI  20
      COMMON ASTER(30000)                                               WOWI  21
      COMMON/PAREM/XB(3),WB(3),IR                                       WOWI  22
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 WOWI  23
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     WOWI  24
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              WOWI  25
      EQUIVALENCE(ASTER,MASTER)                                         WOWI  26
C                                                                       WOWI  27
  901 FORMAT(1H0,32HERROR IN G1 AT 140      BAD ITYPE,5X,4HITY=,I5)     WOWI  28
C                                                                       WOWI  29
      LOC=LREGD+JREG-1                                                  WOWI  30
      CALL UN2(LOC,LOCD,NC)                                             WOWI  31
      CALL UN2(LOCD,IOP,NBO)                                            WOWI  32
      N=1                                                               WOWI  33
      IOPER=IOP                                                         WOWI  34
C                                                                       WOWI  35
C     EXAMINE NC CHOICES N=1,NC                                         WOWI  36
C                                                                       WOWI  37
   10 ITEMP=LIO+NBO-1                                                   WOWI  38
      CALL UN3(ITEMP,LRI,LRO,LOOP)                                      WOWI  39
      ITEMP=LBODY+3*(NBO-1)                                             WOWI  40
      CALL UN2(ITEMP,ITYPE,LOCDA)                                       WOWI  41
      IF(LOOP.NE.KLOOP)GOTO 30                                          WOWI  42
C          CONTINUATION OF RAY                                          WOWI  43
      IF(ITYPE.GT.11)GOTO 40                                            WOWI  44
      IJK=LRIN+NBO-1                                                    WOWI  45
      RIN=ASTER(IJK)                                                    WOWI  46
      IJK=LROT+NBO-1                                                    WOWI  47
      ROUT=ASTER(IJK)                                                   WOWI  48
      IF(ITYPE.LT.10)GOTO 310                                           WOWI  49
C          TOR AND ARS                                                  WOWI  50
C          IF DIST 0 ROUT COMPUT RIN/ROUT SET                           WOWI  51
      IF(ROUT.LT.0.)GOTO 400                                            WOWI  52
      IF(DIST.LE.ROUT)GOTO 310                                          WOWI  53
C                                                                       WOWI  54
   30 LRI=1                                                             WOWI  55
      LRO=1                                                             WOWI  56
      ITY=ITYPE+1                                                       WOWI  57
      IF(ITY.GE.1.AND.ITY.LE.12)GOTO 100                                WOWI  58
   40 IERR=IERR+1                                                       WOWI  59
      WRITE (6,901)ITYPE                                                WOWI  60
      RETURN                                                            WOWI  61
C          RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS              WOWI  62
  100 GOTO(110,120,130,140,150,160,170,180,190,200,210,220),ITY         WOWI  63
  110 CALL RPP(NBO)                                                     WOWI  64
      GOTO 300                                                          WOWI  65
  120 CALL BOX                                                          WOWI  66
      GOTO 300                                                          WOWI  67
  130 CALL SPH                                                          WOWI  68
      GOTO 300                                                          WOWI  69
  140 CALL RCC                                                          WOWI  70
      GOTO 300                                                          WOWI  71
  150 CALL REC                                                          WOWI  72
      GOTO 300                                                          WOWI  73
  160 CALL TRC                                                          WOWI  74
      GOTO 300                                                          WOWI  75
  170 CALL ELL                                                          WOWI  76
      GOTO 300                                                          WOWI  77
```

```
  180 CALL RAW                                                          WOWI  78
      GOTO 300                                                          WOWI  79
  190 CALL ARR                                                          WOWI  80
      GOTO 300                                                          WOWI  81
  200 CALL TEC                                                          WOWI  82
      GOTO 300                                                          WOWI  83
  210 CALL TOR                                                          WOWI  84
      GOTO 300                                                          WOWI  85
  220 CALL ARS                                                          WOWI  86
C                                                                       WOWI  87
  300 IJK=LIO+NBO-1                                                     WOWI  88
      MASTER(IJK)=KLOOP+I15*(LRO+64*LRI)                                WOWI  89
C                                                                       WOWI  90
  310 IF(ROUT.LL.0.)GOTO 330                                            WOWI  91
      IF(ABS(RIN-DIST).GT.DIST*1.0E-6)GOTO 320                          WOWI  92
      RIN=DIST                                                          WOWI  93
      GOTO 330                                                          WOWI  94
C                                                                       WOWI  95
  320 IF(ABS(ROUT-DIST).LE.DIST*1.0E-6)ROUT=DIST                        WOWI  96
C                                                                       WOWI  97
  330 IJK=LRIN+NBO-1                                                    WOWI  98
      ASTER(IJK)=RIN                                                    WOWI  99
      IJK=LROT+NBO-1                                                    WOWI 100
      ASTER(IJK)=ROUT                                                   WOWI 101
C                                                                       WOWI 102
C     TEST CONDITIONS FOR XB IN JREG (LTRUE SET=1)                      WOWI 103
C                                                                       WOWI 104
  400 IF(IOPER.GT.4)GOTO 500                                            WOWI 105
C         (+) OPERATOR TEST FOR INSIDE RIN.LE.DIST.LT.ROUT              WOWI 106
      IF(RIN.GT.DIST)GOTO 700                                           WOWI 107
      IF(DIST-ROUT)600,700,700                                          WOWI 108
C         (-) OPERATOR TEST FOR OUTSIDE  DIST.LT.RIN    DIST.GE.ROUT    WOWI 109
  500 IF(ROUT.LE.0.)GOTO 600                                            WOWI 110
      IF(DIST.LT.RIN)GOTO 600                                           WOWI 111
      IF(DIST.EQ.RIN)GOTO 700                                           WOWI 112
      IF(DIST.LT.ROUT)GOTO 700                                          WOWI 113
C         CHECK NEXT BODY IN DESCRIPTION                                WOWI 114
  600 IF(N.GE.NC)GOTO 800                                               WOWI 115
      N=N+1                                                             WOWI 116
      LOCD=LOCD+1                                                       WOWI 117
      CALL UN2(LOCD,IOPER,NBO)                                          WOWI 118
      IF(IOPER.EQ.1.OR.IOPER.EQ.5)GOTO 800                              WOWI 119
      GOTO 10                                                           WOWI 120
C         OR OPERATOR                                                   WOWI 121
  700 IF(IOP.NE.1.AND.IOP.NE.5) RETURN                                  WOWI 122
      IF(N.GE.NC)RETURN                                                 WOWI 123
      N=N+1                                                             WOWI 124
      DO 710 NN=N,NC                                                    WOWI 125
      LOCD=LOCD+1                                                       WOWI 126
      CALL UN2(LOCD,IOPER,NBO)                                          WOWI 127
      IF(IOPER.NE.1.AND.IOPER.NE.5)GOTO 710                             WOWI 128
      N=NN                                                              WOWI 129
      GOTO 10                                                           WOWI 130
  710 CONTINUE                                                          WOWI 131
      RETURN                                                            WOWI 132
C                                                                       WOWI 133
  800 LTRUE=LTRUE+1                                                     WOWI 134
      RETURN                                                            WOWI 135
      END                                                               WOWI 136
C                                                                       WOWI 137
```

```
C
      SUBROUTINE ARB                                                    WOWI 138
      DIMENSION AA(6,4),XP(3)                                           **** 29
      COMMON ASTER(30000)                                               ARB   2
      COMMON/PAREM/XB(3),WB(3),IR                                       ARB   3
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 ARB   4
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     ARB   5
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              ARB   6
C                                                                       ARB   7
      LOC=LOCDA-1                                                       ARB   8
      DO 10 I=1,6                                                       ARB   9
      LOC=LOC+1                                                         ARB  10
      CALL UN2(LOC,LD,LC)                                               ARB  11
      AA(I,1)=ASTER(LC)                                                 ARB  12
      AA(I,2)=ASTER(LC+1)                                               ARB  13
      AA(I,3)=ASTER(LC+2)                                               ARB  14
      AA(I,4)=ASTER(LD)                                                 ARB  15
   10 CONTINUE                                                          ARB  16
      RIN=-PINF                                                         ARB  17
      ROUT=PINF                                                         ARB  18
      LRO=0                                                             ARB  19
      LRI=0                                                             ARB  20
      S1=0.                                                             ARB  21
      S2=0.                                                             ARB  22
      L1=0                                                              ARB  23
      L2=0                                                              ARB  24
      DO 70 I=1,6                                                       ARB  25
      D=AA(I,4)                                                         ARB  26
      SNUM=-D-AA(I,1)*XB(1)-AA(I,2)*XB(2)-AA(I,3)*XB(3)                 ARB  27
      SDEN=AA(I,1)*WB(1)+AA(I,2)*WB(2)+ AA(I,3)*WB(3)                   ARB  28
      IF(SDEN)20,70,30                                                  ARB  29
   20 IF(SNUM)40,70,70                                                  ARB  30
   30 IF(SNUM)70,70,40                                                  ARB  31
   40 S=SNUM/SDEN                                                       ARB  32
      DO 50 K=1,3                                                       ARB  33
      XP(K)=XB(K)+S*WB(K)                                               ARB  34
   50 CONTINUE                                                          ARB  35
      DO 60 J=1,6                                                       ARB  36
      IF(I.EQ.J)GOTO 60                                                 ARB  37
      T=AA(J,1)*XP(1)+AA(J,2)*XP(2)+AA(J,3)*XP(3)+AA(J,4)               ARB  38
      IF(ABS(T).LE.1.0E-6)T=0.                                          ARB  39
      IF(T.LT.0.)GOTO 70                                                ARB  40
   60 CONTINUE                                                          ARB  41
      IF(L1.GT.0)GOTO 65                                                ARB  42
      L1=I                                                              ARB  43
      S1=S                                                              ARB  44
      GOTO 70                                                           ARB  45
   65 IF(ABS(S1-S).GT.1.0E-6)GOTO 100                                   ARB  46
   70 CONTINUE                                                          ARB  47
C                                                                       ARB  48
      IF(L1)200,200,150                                                 ARB  49
  100 S2=S                                                              ARB  50
      L2=I                                                              ARB  51
      IF(ABS(S1-S2).LE.S1*1.0E-5)GOTO 200                               ARB  52
      IF(S1-S2)110,200,120                                              ARB  53
  110 RIN=S1                                                            ARB  54
      ROUT=S2                                                           ARB  55
      LRI=L1                                                            ARB  56
      LRO=L2                                                            ARB  57
      RETURN                                                            ARB  58
                                                                        ARB  59
```

```
  120 RIN=S2                                                            ARB   60
      LRI=L2                                                            ARB   61
  130 ROUT=S1                                                           ARB   62
      LRO=L1                                                            ARB   63
      RETURN                                                            ARB   64
  150 DO 160 J=1,6                                                      ARB   65
      IF(L1.EQ.J)GOTO 160                                               ARB   66
      T1=AA(J,1)*XB(1)+AA(J,2)*XB(2)+AA(J,3)*XB(3)+AA(J,4)              ARB   67
      IF(ABS(T1).LE.1.0E-6)T1=0.                                        ARB   68
      IF(T1.LT.0.)GOTO 200                                              ARB   69
  160 CONTINUE                                                          ARB   70
      GOTO 130                                                          ARB   71
C                                                                       ARB   72
  200 RIN=PINF                                                          ARB   73
      ROUT=-PINF                                                        ARB   74
      LRI=0                                                             ARB   75
      LRO=0                                                             ARB   76
      RETURN                                                            ARB   77
      END                                                               ARB   78
C                                                                       ARB   79
C                                                                       ARB   80
      SUBROUTINE ARS                                                    ****  30
      DIMENSION MASTER(30000),COL1(3),COL2(3),COL3(3),COL4(3),          ARS    2
     1  U(3),V(3),W(3),SAVE(84)                                         ARS    3
      COMMON ASTER(30000)                                               ARS    4
      COMMON/PAREM/XB(3),WB(3),IR                                       ARS    5
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 ARS    6
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     ARS    7
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              ARS    8
      COMMON/DAVIS/IGRID,LOOP,INORM                                     ARS    9
      EQUIVALENCE(COL11,COL1(1)),(COL12,COL1(2)),(COL13,COL1(3))        ARS   10
      EQUIVALENCE(COL21,COL2(1)),(COL22,COL2(2)),(COL23,COL2(3))        ARS   11
      EQUIVALENCE(COL31,COL3(1)),(COL32,COL3(2)),(COL33,COL3(3))        ARS   12
      EQUIVALENCE(COL41,COL4(1)),(COL42,COL4(2)),(COL43,COL4(3))        ARS   13
      EQUIVALENCE(ASTER,MASTER)                                         ARS   14
C                                                                       ARS   15
  901 FORMAT(1H0,21HTROUBLE IN ARS AT 150)                              ARS   16
  902 FORMAT(1H0,48HPOSSIBLE ERROR IN ARBITRARY SURFACE, CHECK INPUT,I5)ARS   17
C                                                                       ARS   18
C                ASTER/MASTER                                           ARS   19
C     LOCDA+0        T                    TEMPORARY STORAGE             ARS   20
C          +1        M                    NO. CURVES                    ARS   21
C          +2        N                    NO. POINTS/CURVE              ARS   22
C          +3      IGDTL                  GRID TOLERANCE                ARS   23
C          +4      BIAS                   NO. OF NEGATIVE OR ZERO HITS  ARS   24
C          +5      XB(X)                                                ARS   25
C          +6      XB(Y)                                                ARS   26
C          +7      XB(Z)                                                ARS   27
C          +8      (84 WORDS)             RESERVED FOR HITS             ARS   28
C           .        .                                                  ARS   29
C           .        .                                                  ARS   30
C           .        .                                                  ARS   31
C         +91        .                                                  ARS   32
C         +92        X  )      )                                        ARS   33
C           .        Y   )N=1  )          M SETS OF N POINTS            ARS   34
C           .        Z   )       )M=1                                   ARS   35
C           .        K  )        )                                      ARS   36
C           .        .  )        )                                      ARS   37
C           .        .   )N=2    )                                      ARS   38
C           .        .   )       )                                      ARS   39
```

```
C           .         .  )      )                                    ARS   40
C                                                                     ARS   41
C     M = THE NUMBER OF CURVES INPUT                                  ARS   42
C     N = THE NUMBER POINTS/CURVE                                     ARS   43
C     IGDTL = THE NUMBER OF GRID SQUARES TO ALOW AROUND EACH POINT    ARS   44
C     IBIAS = THE INDEX INTO MASTER TO THE NUMBER OF DISCARDED HITS   ARS   45
C                                                                     ARS   46
C     LOCHTS IS THE LOCATION OF THE AREA IN MASTER-ASTER              ARS   47
C         RESERVED FOR STORING HITS                                   ARS   48
C                                                                     ARS   49
C     LOCARY IS THE LOCATION IN MASTER-ASTER OF THE DATA POINTS       ARS   50
C         THEMSELVES IN FORMAT( X , Y , Z , GRID SQUARE)              ARS   51
C                                                                     ARS   52
C                                                                     ARS   53
      NE=4                                                            ARS   54
      IWH=MASTER(LOCDA)                                               ARS   55
      IBIAS=IWH+4                                                     ARS   56
      LOCHTS=IBIAS+4                                                  ARS   57
      IF(INORM.EQ.0)GOTO 20                                           ARS   58
      DO 10 I=1,84                                                    ARS   59
      IJK=LOCHTS+I-1                                                  ARS   60
      SAVE(I)=ASTER(IJK)                                              ARS   61
   10 CONTINUE                                                        ARS   62
      GOTO 30                                                         ARS   63
   20 ASTER(IWH+5)=XB(1)                                              ARS   64
      ASTER(IWH+6)=XB(2)                                              ARS   65
      ASTER(IWH+7)=XB(3)                                              ARS   66
   30 IF(KLOOP.EQ.LOOP)GOTO 400                                       ARS   67
C                                                                     ARS   68
      LRI=1                                                           ARS   69
      LRO=1                                                           ARS   70
      M=MASTER(IWH+1)                                                 ARS   71
      N=MASTER(IWH+2)                                                 ARS   72
      IGDTL=MASTER(IWH+3)                                             ARS   73
      LOCARY=LOCHTS+21*NE                                             ARS   74
      NHITS=0                                                         ARS   75
      ASTER(LOCHTS)=PINF                                              ARS   76
      MASTER(IBIAS)=0                                                 ARS   77
      KAPPA=M-1                                                       ARS   78
C                                                                     ARS   79
C                                                                     ARS   80
C     IN THE EVEN CASE, THE TRIANGLES ARE -                           ARS   81
C                                                                     ARS   82
C     (1)  (I,J)  (I+1,J)  (I,J+1)                                    ARS   83
C     (2)                  (I,J-1)                                    ARS   84
C     (3)         (I-1,J)  (I,J+1)                                    ARS   85
C     (4)                  (I,J-1)                                    ARS   86
C                                                                     ARS   87
C                                                                     ARS   88
C     IN THE ODD CASE, THE TRIANGLES ARE -                            ARS   89
C                                                                     ARS   90
C     (1)  (I,J)  (I+1,J)  (I+1,J+1)                                  ARS   91
C     (2)                  (I+1,J-1)                                  ARS   92
C     (3)         (I-1,J)  (I+1,J+1)                                  ARS   93
C     (4)                  (I+1,J-1)                                  ARS   94
C                                                                     ARS   95
C     NOTE THAT THE ONLY DIFFERENCE IS THE ROW DESIGNATION OF W       ARS   96
C                                                                     ARS   97
C     BECAUSE OF THE INCREMENTATION OF I AND J, WE NEED               ARS   98
C         CONSIDER ONLY CASES (1) AND (2)                             ARS   99
```

```
C
      DO 200 I=1,KAPPA
      DO 200 J=1,N
      ITRY=0
      K=(I+J)/2
      IODD=I+J-2*K
C
      IW1=4*((I-1+IODD)*N+J)+LOCARY
      IF(N.LE.J)GOTO 190
  100 IV1=4*(I*N+(J-1))+LOCARY
      IU1=4*((I-1)*N+(J-1))+LOCARY
      IF(INORM.EQ.0)GOTO 110
      IF(IABS(IGRID-MASTER(IU1+3)).GT.IGDTL)GOTO 200
      IF(IABS(IGRID-MASTER(IV1+3)).GT.IGDTL)GOTO 200
      IF(IABS(IGRID-MASTER(IW1+3)).GT.IGDTL)GOTO 180
  110 DO 115 K=1,3
      IJK=IU1+K-1
      U(K)=ASTER(IJK)
      IJK=IV1+K-1
      V(K)=ASTER(IJK)
      IJK=IW1+K-1
      W(K)=ASTER(IJK)
  115 CONTINUE
C
C     AT THIS TIME WE HAVE U,V,W   SIDES OF TRIANGLE
C
      DO 120 K=1,3
      COL1(K)=U(K)-W(K)
      COL2(K)=V(K)-W(K)
      COL3(K)=-WB(K)
      COL4(K)=XB(K)-W(K)
  120 CONTINUE
      D      = COL11*(COL22*COL33-COL23*COL32)
     1        -COL12*(COL21*COL33-COL23*COL31)
     2        +COL13*(COL21*COL32-COL22*COL31)
      IF(ABS(D).LE.1.0E-6)GOTO 180
C
      DALPHA= COL41*(COL22*COL33-COL23*COL32)
     1        -COL42*(COL21*COL33-COL23*COL31)
     2        +COL43*(COL21*COL32-COL22*COL31)
      ALPHA=DALPHA/D
      IF(ALPHA*(1.-ALPHA).LT.0.)GOTO 180
C
      DBETA = COL11*(COL42*COL33-COL43*COL32)
     1        -COL12*(COL41*COL33-COL43*COL31)
     2        +COL13*(COL41*COL32-COL42*COL31)
      BETA=DBETA/D
      IF(BETA*(1.-BETA).LT.0.)GOTO 180
      TP=ALPHA+BETA
      IF(TP*(1.-TP).LT.0.)GOTO 180
C
      DS     = COL11*(COL22*COL43-COL23*COL42)
     1        -COL12*(COL21*COL43-COL23*COL41)
     2        +COL13*(COL21*COL42-COL22*COL41)
      S=DS/D
C
      IF(NHITS.GT.20)GOTO 400
      LIMIT=NHITS+1
      LIMIT1=LOCHTS+20*NE-1
      TRY=1-ITRY-ITRY
```

```
      CALL CROSS(COL3,COL1,COL2)                                        ARS  160
      DO 140 L=1,3                                                      ARS  161
      COL3(L)=TRY*COL3(L)                                               ARS  162
  140 CONTINUE                                                          ARS  163
      DO 150 L=1,LIMIT                                                  ARS  164
      INDEX=LOCHTS+(L-1)*NE                                             ARS  165
      IF(S.LE.ASTER(INDEX))GOTO 160                                     ARS  166
  150 CONTINUE                                                          ARS  167
      WRITE (6,901)                                                     ARS  168
      GOTO 180                                                          ARS  169
C                                                                       ARS  170
C                                                                       ARS  171
  160 DO 165 L=INDEX,LIMIT1                                             ARS  172
      IJK=LIMIT1+INDEX-L                                                ARS  173
      IJK1=IJK+NE                                                       ARS  174
      ASTER(IJK1)=ASTER(IJK)                                            ARS  175
  165 CONTINUE                                                          ARS  176
      ASTER(INDEX)=S                                                    ARS  177
      DO 170 L=1,3                                                      ARS  178
      IJK=INDEX+L                                                       ARS  179
      ASTER(IJK)=COL3(L)                                                ARS  180
  170 CONTINUE                                                          ARS  181
      NHITS=NHITS+1                                                     ARS  182
  180 IF(ITRY.GT.0)GOTO 200                                             ARS  183
  190 IW1=IW1-8                                                         ARS  184
      ITRY=1                                                            ARS  185
      IF(J.GT.1)GOTO 100                                                ARS  186
  200 CONTINUE                                                          ARS  187
C                                                                       ARS  188
C     THIS SECTION CHECKS FOR PROPER ENTER-LEAVE SEQUENCE IN HITS TABLE ARS  189
C                                                                       ARS  190
      IF(NHITS-1)800,210,220                                            ARS  191
  210 ASTER(LOCHTS)=PINF                                                ARS  192
      IJK=LOCHTS+NE                                                     ARS  193
      ASTER(IJK)=PINF                                                   ARS  194
      GOTO 800                                                          ARS  195
  220 ILEAVE=1                                                          ARS  196
      SLAST=-PINF                                                       ARS  197
C                                                                       ARS  198
C     ILEAVE = -1 IMPLIES AN ENTRY                                      ARS  199
C     ILEAVE = +1 IMPLIES AN EXIT                                       ARS  200
C     ENTRIES AND EXITS SHOULD ALTERNATE IN TABLE                       ARS  201
C                                                                       ARS  202
      DO 300 L=1,NHITS                                                  ARS  203
      INDEX=LOCHTS+(L-1)*NE                                             ARS  204
      DO 230 L1=1,3                                                     ARS  205
      IJK=INDEX+L1                                                      ARS  206
      COL4(L1)=ASTER(IJK)                                               ARS  207
  230 CONTINUE                                                          ARS  208
      TEMP=DOT(WB,COL4)                                                 ARS  209
      INEXT=SIGN(1.0,TEMP)                                              ARS  210
      IF(ABS(SLAST-ASTER(INDEX)).GT.1.0E-7)GOTO 235                     ARS  211
      IF(ILEAVE*INEXT.GE.0)GOTO 260                                     ARS  212
      LTRY=L                                                            ARS  213
      INDEX=INDEX-NE                                                    ARS  214
      GOTO 270                                                          ARS  215
  235 IJK=INDEX+NE                                                      ARS  216
      IF(ABS(ASTER(INDEX)-ASTER(IJK)).GT.1.0E-7)GOTO 240                ARS  217
      IF(ILEAVE*INEXT)290,250,250                                       ARS  218
  240 IF(ILEAVE*INEXT)290,280,280                                       ARS  219
```

```
C
C     BAD START OF A NEW S SET - TRY TO FIND AN ALTERNATING MEMBER
C
  250 LTRY=L
  251 LTRY=LTRY+1
      IF(LTRY.GT.NHITS)GOTO 280
      INDEX1=LOCHTS+(LTRY-1)*NE
      IF(ABS(ASTER(INDEX)-ASTER(INDEX1)).GT.1.0E-7)GOTO 280
      DO 252 L1=1,3
      IJK=INDEX1+L1
      COL4(L1)=ASTER(IJK)
  252 CONTINUE
      TEMP=DOT(WB,COL4)
      INEXT=SIGN(1.0,TEMP)
      IF(ILEAVE*INEXT.GE.0)GOTO 251
      LTRY=L+1
      GOTO 270
C
C     AT THIS POINT WE HAVE DETECTED TWO CONSECUTIVE ENTRIES OR EXITS
C         TRY TO RESOLVE BY DELETING ITEMS WITH EQUAL S ENTRIES
C
  260 LTRY=L
  261 LTRY=LTRY+1
      IF(LTRY.LE.NHITS)GOTO 262
      LTRY=L+1
      GOTO 270
  262 INDEX1=LOCHTS+(LTRY-1)*NE
      IF(ABS(ASTER(INDEX)-ASTER(INDEX1)).LE.1.0E-7)GOTO 263
      LTRY=L+1
      GOTO 270
  263 DO 264 L1=1,3
      IJK=INDEX1+L1
      COL4(L1)=ASTER(IJK)
  264 CONTINUE
      TEMP=DOT(WB,COL4)
      INEXT=SIGN(1.0,TEMP)
      IF(ILEAVE*INEXT.GE.0)GOTO 261
      LTRY=L
      INDEX=INDEX-NE
      GOTO 270
C
C      PROCEED TO FORGET FROM INDEX THRU NEXT ENTRY WITH DIFFERENT S
C         COMMENCING TO CHECK WITH THE L TH ENTRY
C
  273 INDEX1=LOCHTS+(LTRY-1)*NE
      IF(ABS(ASTER(INDEX1)-ASTER(INDEX)).GT.0.)GOTO 271
      LTRY=LTRY+1
  270 NHITS=NHITS-1
      IF(NHITS-1)800,210,273
  271 DO 272 LTRY=INDEX,LIMIT1
      IJK=LTRY+INDEX1-INDEX
      ASTER(LTRY)=ASTER(IJK)
  272 CONTINUE
      GOTO 220
C
  280 WRITE (6,902)INDEX
  290 SLAST=ASTER(INDEX)
      ILEAVE=INEXT
  300 CONTINUE
```

```
C     NOW CHOOSE THE HIT (THIS SECTION ALSO ENTERED FOR REENTRY)
C
  400 DO 420 I=1,20
      I1ST=LOCHTS+(I-1)*NE
      I2ND=LOCHTS+I*NE
      IF(ASTER(I2ND).GE.PINF)GOTO 800
      IF(ASTER(I1ST).GE.PINF)GOTO 800
      IF(ABS(ASTER(I1ST)-ASTER(I2ND)).LE.1.0E-7)GOTO 420
      IF(DIST.LT.ASTER(I1ST))GOTO 410
      IF(DIST.GT.ASTER(I2ND))GOTO 420
  410 K=(MASTER(IBIAS)+I)/2
      IF(2*K-I-MASTER(IBIAS))500,510,510
  420 CONTINUE
C
  500 RIN=ASTER(I1ST)
      ROUT=ASTER(I2ND)
      GOTO 810
  510 RIN=ASTER(I2ND)
      IJK=I2ND+NE
      ROUT=ASTER(IJK)
      GOTO 810
  800 RIN=-PINF
      ROUT=0.
  810 IF(NASC.GT.-2)RETURN
      DO 820 I=1,84
      IJK=LOCHTS+I-1
      ASTER(IJK)=SAVE(I)
  820 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE BOX
      DIMENSION MASTER(30000)
      COMMON ASTER(30000)
      COMMON/PAREM/XB(3),WB(3),IR
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
      EQUIVALENCE (MASTER,ASTER)
C
      CALL UN2(LOCDA,IV,IH1)
      LOC=LOCDA+1
      CALL UN2(LOC,IH2,IH3)
      RIN=-PINF
      ROUT=PINF
      DO 105 I=1,3
      IF(I-2)11,12,13
   11 II=2
      GOTO 14
   12 II=1
      GOTO 14
   13 II=3
   14 A=0.
      VP=0.
      W=0.
      DO 15 J=1,3
      JV=IV+J
      JA=IH1+J
      VP=VP+(ASTER(JV-1)-XB(J))*ASTER(JA-1)
```

```
      W=W+WB(J)*ASTER(JA-1)                                             BOX   29
      A=A+ASTER(JA-1)**2                                                BOX   30
   15 CONTINUE                                                          BOX   31
      IF(W)30,20,40                                                     BOX   32
   20 IF(-VP.LT.0.)GOTO 200                                             BOX   33
      IF(-VP-A)100,100,200                                              BOX   34
   30 CP=VP/W                                                           BOX   35
      LO=2*II-1                                                         BOX   36
      IF(CP.LE.0.)GOTO 200                                              BOX   37
      CM=(VP+A)/W                                                       BOX   38
      LI=LO+1                                                           BOX   39
      GOTO 60                                                           BOX   40
   40 CP=(VP+A)/W                                                       BOX   41
      LO=2*II                                                           BOX   42
      IF(CP.LE.0.)GOTO 200                                              BOX   43
      CM=VP/W                                                           BOX   44
      LI=LO-1                                                           BOX   45
   60 IF(ROUT.LE.CP)GOTO 80                                             BOX   46
      ROUT=CP                                                           BOX   47
      LRO=LO                                                            BOX   48
   80 IF(RIN.GE.CM)GOTO 100                                             BOX   49
      RIN=CM                                                            BOX   50
      LRI=LI                                                            BOX   51
  100 IH1=IH2                                                           BOX   52
      IH2=IH3                                                           BOX   53
  105 CONTINUE                                                          BOX   54
      IF(ABS(RIN-ROUT).LE.ROUT*1.0E-6)GOTO 200                          BOX   55
      IF(RIN.LT.ROUT)RETURN                                             BOX   56
  200 RIN=PINF                                                          BOX   57
      ROUT=-PINF                                                        BOX   58
      RETURN                                                            BOX   59
      END                                                               BOX   60
C                                                                       BOX   61
C                                                                       BOX   62
      SUBROUTINE ELL                                                    ****  32
      DIMENSION FOCIA(3),FOCIB(3),MASTER(30000)                         ELL    2
      COMMON ASTER(30000)                                               ELL    3
      COMMON/PAREM/XB(3),WB(3),IR                                       ELL    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 ELL    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     ELL    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              ELL    7
      EQUIVALENCE (ASTER,MASTER)                                        ELL    8
C                                                                       ELL    9
      CALL UN2(LOCDA,IV1,IV2)                                           ELL   10
      IRR=MASTER(LOCDA+1)                                               ELL   11
      FOCIA(1)=ASTER(IV1)                                               ELL   12
      FOCIA(2)=ASTER(IV1+1)                                             ELL   13
      FOCIA(3)=ASTER(IV1+2)                                             ELL   14
      FOCIB(1)=ASTER(IV2)                                               ELL   15
      FOCIB(2)=ASTER(IV2+1)                                             ELL   16
      FOCIB(3)=ASTER(IV2+2)                                             ELL   17
      C=ASTER(IRR)                                                      ELL   18
      RIN=PINF                                                          ELL   19
      ROUT=-PINF                                                        ELL   20
      D1X=XB(1)-FOCIA(1)                                                ELL   21
      D1Y=XB(2)-FOCIA(2)                                                ELL   22
      D1Z=XB(3)-FOCIA(3)                                                ELL   23
      D2X=XB(1)-FOCIB(1)                                                ELL   24
      D2Y=XB(2)-FOCIB(2)                                                ELL   25
      D2Z=XB(3)-FOCIB(3)                                                ELL   26
```

```
      A1=2.*(D1X*WB(1)+D1Y*WB(2)+D1Z*WB(3))                             ELL   27
      A2=2.*(D2X*WB(1)+D2Y*WB(2)+D2Z*WB(3))                             ELL   28
      B1=D1X*D1X+D1Y*D1Y+D1Z*D1Z                                        ELL   29
      B2=D2X*D2X+D2Y*D2Y+D2Z*D2Z                                        ELL   30
      AA=(A2-A1)/(2.*C)                                                 ELL   31
      BB=(C*C+B2-B1)/(2.*C)                                             ELL   32
      ALAMD=AA*AA-1.                                                    ELL   33
      ALAM1=(AA*BB-.5*A2)/ALAMD                                         ELL   34
      U=(BB*BB-B2)/ALAMD                                                ELL   35
      DISCRM=ALAM1*ALAM1-U                                              ELL   36
      IF(DISCRM.LE.0.)RETURN                                            ELL   37
      SQRTDI=SQRT(DISCRM)                                               ELL   38
      RIN=-ALAM1-SQRTDI                                                 ELL   39
      ROUT=-ALAM1+SQRTDI                                                ELL   40
      RETURN                                                            ELL   41
      END                                                               ELL   42
C                                                                       ELL   43
C                                                                       ELL   44
      SUBROUTINE RAW                                                    ****  33
      DIMENSION H1(3),H2(3),H3(3),V(3),ASQ(3),PV(4),G(3)                RAW    2
      COMMON ASTER(30000)                                               RAW    3
      COMMON/PAREM/XB(3),WB(3),IR                                       RAW    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 RAW    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     RAW    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              RAW    7
C                                                                       RAW    8
      CALL UN2(LOCDA,IV,IH1)                                            RAW    9
      LOC=LOCDA+1                                                       RAW   10
      CALL UN2(LOC,IH2,IH3)                                             RAW   11
      H1(1)=ASTER(IH1)                                                  RAW   12
      H1(2)=ASTER(IH1+1)                                                RAW   13
      H1(3)=ASTER(IH1+2)                                                RAW   14
      H2(1)=ASTER(IH2)                                                  RAW   15
      H2(2)=ASTER(IH2+1)                                                RAW   16
      H2(3)=ASTER(IH2+2)                                                RAW   17
      H3(1)=ASTER(IH3)                                                  RAW   18
      H3(2)=ASTER(IH3+1)                                                RAW   19
      H3(3)=ASTER(IH3+2)                                                RAW   20
      V(1)=ASTER(IV)                                                    RAW   21
      V(2)=ASTER(IV+1)                                                  RAW   22
      V(3)=ASTER(IV+2)                                                  RAW   23
      RIN=-PINF                                                         RAW   24
      ROUT=PINF                                                         RAW   25
      CM=-PINF                                                          RAW   26
      CP=PINF                                                           RAW   27
      L=0                                                               RAW   28
      L1=0                                                              RAW   29
      K=0                                                               RAW   30
      LRI=0                                                             RAW   31
      LRO=0                                                             RAW   32
      ASQ(1)=H1(1)*H1(1)+H1(2)*H1(2)+H1(3)*H1(3)                        RAW   33
      ASQ(2)=H2(1)*H2(1)+H2(2)*H2(2)+H2(3)*H2(3)                        RAW   34
      ASQ(3)=H3(1)*H3(1)+H3(2)*H3(2)+H3(3)*H3(3)                        RAW   35
      XB1V1=XB(1)-V(1)                                                  RAW   36
      XB2V2=XB(2)-V(2)                                                  RAW   37
      XB3V3=XB(3)-V(3)                                                  RAW   38
      PV(1)=XB1V1*H1(1)+XB2V2*H1(2)+XB3V3*H1(3)                         RAW   39
      PV(2)=XB1V1*H2(1)+XB2V2*H2(2)+XB3V3*H2(3)                         RAW   40
      PV(3)=XB1V1*H3(1)+XB2V2*H3(2)+XB3V3*H3(3)                         RAW   41
      G(1)=WB(1)*H1(1)+WB(2)*H1(2)+WB(3)*H1(3)                          RAW   42
```

```
      G(2)=WB(1)*H2(1)+WB(2)*H2(2)+WB(3)*H2(3)                          RAW   43
      G(3)=WB(1)*H3(1)+WB(2)*H3(2)+WB(3)*H3(3)                          RAW   44
C                                                                       RAW   45
      DO 140 I=1,2                                                      RAW   46
      IF(G(I))10,110,60                                                 RAW   47
C                                                                       RAW   48
   10 IF(-PV(I))20,400,400                                              RAW   49
   20 TEMP=-PV(I)/G(I)                                                  RAW   50
      IF(TEMP-CP)30,130,130                                             RAW   51
   30 CP=TEMP                                                           RAW   52
      L=I                                                               RAW   53
      GOTO(40,50),I                                                     RAW   54
   40 LRO=3                                                             RAW   55
      GOTO 130                                                          RAW   56
   50 LRO=1                                                             RAW   57
      GOTO 130                                                          RAW   58
C                                                                       RAW   59
   60 IF(-PV(I).LE.0.)GOTO 130                                          RAW   60
      TEMP=-PV(I)/G(I)                                                  RAW   61
      IF(TEMP.LE.CM)GOTO 130                                            RAW   62
      CM=TEMP                                                           RAW   63
      K=I                                                               RAW   64
      GOTO(90,100),I                                                    RAW   65
   90 LRI=3                                                             RAW   66
      GOTO 130                                                          RAW   67
  100 LRI=1                                                             RAW   68
      GOTO 130                                                          RAW   69
C                                                                       RAW   70
  110 IF(PV(I).LE.0.)GOTO 810                                           RAW   71
      IF(PV(I).GE.ASQ(I))GOTO 810                                       RAW   72
  130 L1=L1+I                                                           RAW   73
  140 CONTINUE                                                          RAW   74
C                                                                       RAW   75
      IF(G(3))150,210,230                                               RAW   76
  150 TEMP=ASQ(3)-PV(3)                                                 RAW   77
      IF(TEMP.GE.0.)GOTO 180                                            RAW   78
      TEMP=TEMP/G(3)                                                    RAW   79
      IF(TEMP.LE.CM)GOTO 190                                            RAW   80
      CM=TEMP                                                           RAW   81
      K=3                                                               RAW   82
      LRI=6                                                             RAW   83
  180 IF(-PV(3))190,400,400                                             RAW   84
  190 TEMP=-PV(3)/G(3)                                                  RAW   85
      IF(TEMP.GE.CP)GOTO 290                                            RAW   86
      CP=TEMP                                                           RAW   87
      L=3                                                               RAW   88
      LRO=5                                                             RAW   89
      GOTO 290                                                          RAW   90
C.                                                                      RAW   91
  210 IF(PV(3).LE.0.)GOTO 400                                           RAW   92
      IF(PV(3)-ASQ(3))290,290,400                                       RAW   93
C                                                                       RAW   94
  230 IF(-PV(3).LE.0.)GOTO 260                                          RAW   95
      TEMP=-PV(3)/G(3)                                                  RAW   96
      IF(TEMP.LE.CM)GOTO 260                                            RAW   97
      CM=TEMP                                                           RAW   98
      K=3                                                               RAW   99
      LRI=5                                                             RAW  100
  260 TEMP=ASQ(3)-PV(3)                                                 RAW  101
      IF(TEMP.LE.0.)GOTO 400                                            RAW  102
```

```
      TEMP=TEMP/G(3)                                                    RAW  103
      IF(TEMP.GE.CP)GOTO 290                                            RAW  104
      CP=TEMP                                                           RAW  105
      L=3                                                               RAW  106
      LRO=6                                                             RAW  107
  290 AG=ASQ(2)*G(1)+ASQ(1)*G(2)                                        RAW  108
      PV(4)=PV(1)*ASQ(2)+PV(2)*ASQ(1)                                   RAW  109
      TOP=ASQ(1)*ASQ(2)-PV(4)                                           RAW  110
      IF(AG)310,350,330                                                 RAW  111
  310 TEMP=TOP/AG                                                       RAW  112
      IF(TEMP.LE.CM)GOTO 380                                            RAW  113
      CM=TEMP                                                           RAW  114
      K=4                                                               RAW  115
      LRI=2                                                             RAW  116
      GOTO 380                                                          RAW  117
C                                                                       RAW  118
  330 IF(TOP.LT.0.)GOTO 400                                             RAW  119
      TEMP=TOP/AG                                                       RAW  120
      IF(TEMP-CP)370,380,380                                            RAW  121
C                                                                       RAW  122
  350 IF(PV(4).LE.0.)GOTO 400                                           RAW  123
      IF(-TOP)380,400,400                                               RAW  124
  370 CP=TEMP                                                           RAW  125
      L=4                                                               RAW  126
      LRO=2                                                             RAW  127
  380 IF(L+K.LE.0)GOTO 400                                              RAW  128
      ROUT=CP                                                           RAW  129
      RIN=CM                                                            RAW  130
C                                                                       RAW  131
  400 IF(ROUT.GE.PINF)GOTO 810                                          RAW  132
      IF(ROUT.LE.0.)GOTO 810                                            RAW  133
      IF(RIN.GE.ROUT)GOTO 810                                           RAW  134
      IF(ABS(RIN-ROUT).GT.ROUT*1.0E-5)GOTO 820                          RAW  135
C                                                                       RAW  136
  810 ROUT=-PINF                                                        RAW  137
      RIN=PINF                                                          RAW  138
      LRO=0                                                             RAW  139
      LRI=0                                                             RAW  140
  820 RETURN                                                            RAW  141
      END                                                               RAW  142
C                                                                       RAW  143
C                                                                       RAW  144
      SUBROUTINE RCC                                                    ****  34
      DIMENSION V(3),H(3),MASTER(30000)                                 RCC    2
      COMMON ASTER(30000)                                               RCC    3
      COMMON/PAREM/XB(3),WB(3),IR                                       RCC    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 RCC    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     RCC    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              RCC    7
      EQUIVALENCE (ASTER,MASTER)                                        RCC    8
C                                                                       RCC    9
      CALL UN2(LOCDA,IV,IH)                                             RCC   10
      IRR=MASTER(LOCDA+1)                                               RCC   11
      H(1)=ASTER(IH)                                                    RCC   12
      H(2)=ASTER(IH+1)                                                  RCC   13
      H(3)=ASTER(IH+2)                                                  RCC   14
      V(1)=ASTER(IV)                                                    RCC   15
      V(2)=ASTER(IV+1)                                                  RCC   16
      V(3)=ASTER(IV+2)                                                  RCC   17
      R=ASTER(IRR)                                                      RCC   18
```

```
      RIN=-PINF                                                         RCC   19
      ROUT=PINF                                                         RCC   20
      RSQ=R*R                                                           RCC   21
      LRO=0                                                             RCC   22
      LRI=0                                                             RCC   23
      TOP=0.                                                            RCC   24
      POT=0.                                                            RCC   25
      HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)                                  RCC   26
      VPH=H(1)*(V(1)-XB(1))+H(2)*(V(2)-XB(2))+H(3)*(V(3)-XB(3))         RCC   27
      WH=WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)                               RCC   28
      DEN=HH-WH*WH                                                      RCC   29
      DO 10 I=1,3                                                       RCC   30
      TOP=TOP+WB(I)*(XB(I)-V(I))                                        RCC   31
      POT=POT+(XB(I)-V(I))**2                                           RCC   32
   10 CONTINUE                                                          RCC   33
      AMBD=-HH*TOP-WH*VPH                                               RCC   34
      UM=(POT-RSQ)*HH-VPH**2                                            RCC   35
      IF(WH)40,70,50                                                    RCC   36
   40 CP=VPH/WH                                                         RCC   37
      CM=(VPH+HH)/WH                                                    RCC   38
      LCP=1                                                             RCC   39
      LCM=2                                                             RCC   40
      GOTO 60                                                           RCC   41
   50 CP=(VPH+HH)/WH                                                    RCC   42
      CM=VPH/WH                                                         RCC   43
      LCM=1                                                             RCC   44
      LCP=2                                                             RCC   45
   60 IF(CP)300,80,80                                                   RCC   46
   70 CP=PINF                                                           RCC   47
      CM=-CP                                                            RCC   48
      IF(VPH.GT.0.)GOTO 300                                             RCC   49
      IF(HH+VPH)300,90,90                                               RCC   50
   80 IF(ABS(DEN).GE.1.0E-6)GOTO 90                                     RCC   51
      R1=-PINF                                                          RCC   52
      R2=PINF                                                           RCC   53
      GOTO 100                                                          RCC   54
   90 R1=0.                                                             RCC   55
      R2=0.                                                             RCC   56
      AMBDA=AMBD/DEN                                                    RCC   57
      UMU=UM/DEN                                                        RCC   58
      DISC=AMBDA**2-UMU                                                 RCC   59
      IF(DISC.LE.0.)GOTO 300                                            RCC   60
      SD=SQRT(DISC)                                                     RCC   61
      R1=AMBDA-SD                                                       RCC   62
      R2=AMBDA+SD                                                       RCC   63
  100 IF(CM.GT.R1)GOTO 110                                              RCC   64
      RIN=R1                                                            RCC   65
      LRI=3                                                             RCC   66
      GOTO 120                                                          RCC   67
  110 RIN=CM                                                            RCC   68
      LRI=LCM                                                           RCC   69
  120 IF(CP.LE.R2)GOTO 130                                              RCC   70
      ROUT=R2                                                           RCC   71
      LRO=3                                                             RCC   72
      GOTO 200                                                          RCC   73
  130 ROUT=CP                                                           RCC   74
      LRO=LCP                                                           RCC   75
  200 IF(ABS(ROUT-RIN).LE.ROUT*1.0E-5)GOTO 300                          RCC   76
      GOTO(210,210,220),LRO                                             RCC   77
  210 F1=DEN*ROUT**2-2.*AMBD*ROUT+UM                                    RCC   78
```

```
      IF(F1)250,250,300                                                 RCC   79
  220 F1=ROUT*WH-VPH                                                    RCC   80
      IF(F1)300,250,230                                                 RCC   81
      GOTO 230                                                          RCC   82
  230 IF(F1.GT.HH) GOTO 300                                             RCC   83
  250 GOTO(260,260,270),LRI                                             RCC   84
  260 F1=DEN*RIN**2-2.*AMBD*RIN+UM                                      RCC   85
      IF(F1)310,310,300                                                 RCC   86
  270 F1=RIN*WH-VPH                                                     RCC   87
      IF(F1)300,310,280                                                 RCC   88
      GOTO 280                                                          RCC   89
  280 IF(F1.LE.HH)GOTO 310                                              RCC   90
  300 RIN=PINF                                                          RCC   91
      ROUT=-PINF                                                        RCC   92
      LRO=0                                                             RCC   93
      LRI=0                                                             RCC   94
  310 RETURN                                                            RCC   95
      END                                                               RCC   96
C                                                                       RCC   97
C                                                                       RCC   98
      SUBROUTINE REC                                                    ****  35
      DIMENSION V(3),H(3),A(3),B(3)                                     REC    2
      COMMON ASTER(30000)                                               REC    3
      COMMON/PAREM/XB(3),WB(3),IR                                       REC    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 REC    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     REC    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              REC    7
C                                                                       REC    8
      CALL UN2(LOCDA,IV,IH)                                             REC    9
      LOC=LOCDA+1                                                       REC   10
      CALL UN2(LOC,IA,IB)                                               REC   11
      V(1)=ASTER(IV)                                                    REC   12
      V(2)=ASTER(IV+1)                                                  REC   13
      V(3)=ASTER(IV+2)                                                  REC   14
      H(1)=ASTER(IH)                                                    REC   15
      H(2)=ASTER(IH+1)                                                  REC   16
      H(3)=ASTER(IH+2)                                                  REC   17
      A(1)=ASTER(IA)                                                    REC   18
      A(2)=ASTER(IA+1)                                                  REC   19
      A(3)=ASTER(IA+2)                                                  REC   20
      B(1)=ASTER(IB)                                                    REC   21
      B(2)=ASTER(IB+1)                                                  REC   22
      B(3)=ASTER(IB+2)                                                  REC   23
      RIN=-PINF                                                         REC   24
      ROUT=PINF                                                         REC   25
      LRO=0                                                             REC   26
      LRI=0                                                             REC   27
      AA=A(1)*A(1)+A(2)*A(2)+A(3)*A(3)                                  REC   28
      BB=B(1)*B(1)+B(2)*B(2)+B(3)*B(3)                                  REC   29
      V1XB1=V(1)-XB(1)                                                  REC   30
      V2XB2=V(2)-XB(2)                                                  REC   31
      V3XB3=V(3)-XB(3)                                                  REC   32
      VPA=V1XB1*A(1)+V2XB2*A(2)+V3XB3*A(3)                              REC   33
      VPB=V1XB1*B(1)+V2XB2*B(2)+V3XB3*B(3)                              REC   34
      WBA=WB(1)*A(1)+WB(2)*A(2)+WB(3)*A(3)                              REC   35
      WBB=WB(1)*B(1)+WB(2)*B(2)+WB(3)*B(3)                              REC   36
      WBAWBA=WBA*WBA                                                    REC   37
      WBBWBB=WBB*WBB                                                    REC   38
      AAAA=AA*AA                                                        REC   39
      BBBB=BB*BB                                                        REC   40
```

```
      AMBD=WBA*VPA*BBBB+WBB*VPB*AAAA                                    REC   41
      UM=BBBB*VPA*VPA+AAAA*VPB*VPB-AAAA*BBBB                            REC   42
      DEN=WBAWBA*BBBB+WBBWBB*AAAA                                       REC   43
      IF(ABS(DEN).LE.1.0E-6)GOTO 10                                     REC   44
      AMBDA=AMBD/DEN                                                    REC   45
      UMU=UM/DEN                                                        REC   46
      DISC=AMBDA**2-UMU                                                 REC   47
      IF(DISC.LE.0.)GOTO 300                                            REC   48
      SD=SQRT(DISC)                                                     REC   49
      R1=AMBDA-SD                                                       REC   50
      R2=AMBDA+SD                                                       REC   51
      GOTO 20                                                           REC   52
   10 R1=-PINF                                                          REC   53
      R2=PINF                                                           REC   54
   20 HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)                                  REC   55
      WH=WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)                               REC   56
      VPH=V1XB1*H(1)+V2XB2*H(2)+V3XB3*H(3)                              REC   57
      IF(WH)40,70,50                                                    REC   58
   40 IF(VPH.GE.0.)GOTO 300                                             REC   59
      CP=VPH/WH                                                         REC   60
      CM=(VPH+HH)/WH                                                    REC   61
      LCP=1                                                             REC   62
      LCM=2                                                             REC   63
      GOTO 100                                                          REC   64
   50 VPHHH=VPH+HH                                                      REC   65
      IF(VPHHH.LE.0.)GOTO 300                                           REC   66
      CP=VPHHH/WH                                                       REC   67
      CM=VPH/WH                                                         REC   68
      LCM=1                                                             REC   69
      LCP=2                                                             REC   70
      GOTO 100                                                          REC   71
   70 CP=PINF                                                           REC   72
      CM=-CP                                                            REC   73
  100 IF(CM.GT.R1)GOTO 110                                              REC   74
      RIN=R1                                                            REC   75
      LRI=3                                                             REC   76
      GOTO 120                                                          REC   77
  110 RIN=CM                                                            REC   78
      LRI=LCM                                                           REC   79
  120 IF(CP.LE.R2)GOTO 130                                              REC   80
      ROUT=R2                                                           REC   81
      LRO=3                                                             REC   82
      GOTO 200                                                          REC   83
  130 ROUT=CP                                                           REC   84
      LRO=LCP                                                           REC   85
  200 IF(ABS(ROUT-RIN).LE.ROUT*1.0E-5)GOTO 300                          REC   86
      GOTO(210,210,220),LRO                                             REC   87
  210 F1=DEN*ROUT**2-2.*AMBD*ROUT+UM                                    REC   88
      IF(F1)250,250,300                                                 REC   89
  220 F1=ROUT*WH-VPH                                                    REC   90
      IF(F1)300,250,230                                                 REC   91
      GOTO 230                                                          REC   92
  230 IF(F1.GT.HH)GOTO 300                                              REC   93
  250 GOTO(260,260,270),LRI                                             REC   94
  260 F1=DEN*RIN**2-2.*AMBD*RIN+UM                                      REC   95
      IF(F1)310,310,300                                                 REC   96
  270 F1=RIN*WH-VPH                                                     REC   97
      IF(F1)300,310,280                                                 REC   98
      GOTO 280                                                          REC   99
  280 IF(F1.LE.HH)GOTO 310                                              REC  100
```

```
  300 RIN=PINF                                                          REC
      ROUT=-PINF                                                        REC
      LRI=0                                                             REC
      LRO=0                                                             REC
  310 RETURN                                                            REC
      END                                                               REC
C                                                                       REC
C                                                                       REC
      SUBROUTINE RPP(NBO)                                               ****  36
      DIMENSION MASTER(30000),PR(6),LR(6),XS(6),LST(6)                  RPP    2
      COMMON ASTER(30000)                                               RPP    3
      COMMON/PAREM/XB(3),WB(3),IR                                       RPP    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 RPP    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     RPP    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              RPP    7
      EQUIVALENCE (MASTER,ASTER)                                        RPP    8
C                                                                       RPP    9
  901 FORMAT(1H0,12HERROR IN RPP/4H L =,I10,5X,4HNBO=,I10,5X,3HIR=,     RPP   10
     1  I10/4H XB=,3E20.10/4H WB=,3E20.10/4H PR=,6E20.10/4H LR=,6I10)   RPP   11
C                                                                       RPP   12
      LST(1)=1                                                          RPP   13
      LST(2)=1                                                          RPP   14
      LST(3)=2                                                          RPP   15
      LST(4)=2                                                          RPP   16
      LST(5)=3                                                          RPP   17
      LST(6)=3                                                          RPP   18
      L=0                                                               RPP   19
      PR(1)=0.                                                          RPP   20
      PR(2)=0.                                                          RPP   21
      DO 10 I=1,6                                                       RPP   22
      XS(I)=S(NBO,I)                                                    RPP   23
   10 CONTINUE                                                          RPP   24
C                                                                       RPP   25
      DO 100 I=1,6                                                      RPP   26
      II=LST(I)                                                         RPP   27
      TEMP=XS(I)-XB(II)                                                 RPP   28
      IF(WB(II)) 20,100,30                                              RPP   29
   20 IF(TEMP)40,100,100                                                RPP   30
   30 IF(TEMP.LE.0.)GOTO 100                                            RPP   31
   40 TRY=TEMP/WB(II)                                                   RPP   32
      DO 60 J=1,3                                                       RPP   33
      IF(J.EQ.II)GOTO 60                                                RPP   34
      XRY=XB(J)+TRY*WB(J)                                               RPP   35
      IF((XS(2*J-1)-XRY)*(XRY-XS(2*J)).LT.0.)GOTO 100                   RPP   36
   60 CONTINUE                                                          RPP   37
      L=L+1                                                             RPP   38
      PR(L)=TRY                                                         RPP   39
      LR(L)=I                                                           RPP   40
      IF(L.EQ.2)GOTO 130                                                RPP   41
      IF(L.LT.2)GOTO 100                                                RPP   42
      WRITE (6,901)L,NBO,IR,XB,WB,PR,LR                                 RPP   43
      ROUT=-PINF                                                        RPP   44
      RETURN                                                            RPP   45
  100 CONTINUE                                                          RPP   46
      GOTO 160                                                          RPP   47
C                                                                       RPP   48
  130 IF(ABS(PR(1)-PR(2)).LE.PR(1)*1.0E-6)GOTO 200                      RPP   49
      IF(PR(1)-PR(2))140,180,150                                        RPP   50
  140 RIN=PR(1)                                                         RPP   51
      LRI=LR(1)                                                         RPP   52
```

```
      ROUT=PR(2)                                                        RPP   53
      LRO=LR(2)                                                         RPP   54
      RETURN                                                            RPP   55
  150 RIN=PR(2)                                                         RPP   56
      LRI=LR(2)                                                         RPP   57
      ROUT=PR(1)                                                        RPP   58
      LRO=LR(1)                                                         RPP   59
      RETURN                                                            RPP   60
C                                                                       RPP   61
  160 IF(L.GE.1)GOTO 180                                                RPP   62
  170 ROUT=-PINF                                                        RPP   63
      RETURN                                                            RPP   64
  180 RIN=-PINF                                                         RPP   65
      LRI=0                                                             RPP   66
      ROUT=PR(1)                                                        RPP   67
      LRO=LR(1)                                                         RPP   68
      RETURN                                                            RPP   69
C                                                                       RPP   70
  200 DO 220 J=1,3                                                      RPP   71
      IF(XB(J).LT.XS(2*J-1))GOTO 170                                    RPP   72
      IF(XB(J).GT.XS(2*J))GOTO 170                                      RPP   73
  220 CONTINUE                                                          RPP   74
      GOTO 180                                                          RPP   75
      END                                                               RPP   76
C                                                                       RPP   77
C                                                                       RPP   78
      SUBROUTINE RPP2(LSURF,XP,IRP)                                     ****  37
C         FINDS ABUTING RPP                                             RPP2   2
      DIMENSION XP(3)                                                   RPP2   3
      COMMON ASTER(30000)                                               RPP2   4
      COMMON/PAREM/XB(3),WB(3),IR                                       RPP2   5
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 RPP2   6
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     RPP2   7
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              RPP2   8
C                                                                       RPP2   9
      LOC=LBASE+12*(NASC-1)-2*(LSURF+1)                                 RPP2  10
      CALL UN2(LOC,LOCAT,NC)                                            RPP2  11
      IF(NC-1)10,20,30                                                  RPP2  12
   10 IRP=0                                                             RPP2  13
      RETURN                                                            RPP2  14
   20 CALL UN2(LOCAT,IRP,DUM)                                           RPP2  15
      RETURN                                                            RPP2  16
   30 M=1                                                               RPP2  17
C                                                                       RPP2  18
      DO 90 I=1,NC                                                      RPP2  19
      M=-M                                                              RPP2  20
      IF(M.GT.0)GOTO 50                                                 RPP2  21
      CALL UN2(LOCAT,I1,I2)                                             RPP2  22
      LOCAT=LOCAT+1                                                     RPP2  23
      IRP=I1                                                            RPP2  24
      GOTO 70                                                           RPP2  25
   50 IRP=I2                                                            RPP2  26
   70 LS=(1-LSURF)/2                                                    RPP2  27
      DO 80 J=1,3                                                       RPP2  28
      IF(J.EQ.LS)GOTO 80                                                RPP2  29
      IF((S(IRP,2*J-1)-XP(J))*(XP(J)-S(IRP,2*J)).LT.0.)GOTO 90          RPP2  30
   80 CONTINUE                                                          RPP2  31
      RETURN                                                            RPP2  32
   90 CONTINUE                                                          RPP2  33
      IRP=0                                                             RPP2  34
```

```
      RETURN                                                            RPP2  35
      END                                                               RPP2  36
C                                                                       RPP2  37
C                                                                       RPP2  38
      SUBROUTINE SPH                                                    ****  38
      COMMON ASTER(30000)                                               SPH    2
      COMMON/PAREM/XB(3),WB(3),IR                                       SPH    3
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 SPH    4
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     SPH    5
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              SPH    6
C                                                                       SPH    7
      CALL UN2(LOCDA,ITEMP,I2)                                          SPH    8
      R=ASTER(I2)                                                       SPH    9
      ITEMP=ITEMP+1                                                     SPH   10
      DX=XB(1)-ASTER(ITEMP-1)                                           SPH   11
      DY=XB(2)-ASTER(ITEMP)                                             SPH   12
      DZ=XB(3)-ASTER(ITEMP+1)                                           SPH   13
      B=DX*WB(1)+DY*WB(2)+DZ*WB(3)                                      SPH   14
      C=DX*DX+DY*DY+DZ*DZ-R*R                                           SPH   15
      DIS=B*B-C                                                         SPH   16
      IF(C.GT.0.)GOTO 10                                                SPH   17
      RIN=-PINF                                                         SPH   18
      ROUT=SQRT(DIS)-B                                                  SPH   19
      RETURN                                                            SPH   20
   10 IF(DIS.GT.0.)GOTO 20                                              SPH   21
      RIN=PINF                                                          SPH   22
      ROUT=-PINF                                                        SPH   23
      RETURN                                                            SPH   24
   20 DIS=SQRT(DIS)                                                     SPH   25
      RIN=-B-DIS                                                        SPH   26
      ROUT=-B+DIS                                                       SPH   27
      RETURN                                                            SPH   28
      END                                                               SPH   29
C                                                                       SPH   30
C                                                                       SPH   31
      SUBROUTINE TEC                                                    ****  39
      DIMENSION MASTER(30000),DELTA(3),HF(3),AUN(3)                     TEC    2
      COMMON ASTER(30000)                                               TEC    3
      COMMON/PAREM/XB(3),WB(3),IR                                       TEC    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 TEC    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     TEC    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              TEC    7
      EQUIVALENCE (MASTER,ASTER),(GAMMA,SIGMA)                          TEC    8
      REAL NF(3),K(3),M,M2,MM,MM2                                       TEC    9
C                                                                       TEC   10
      CALL UN2(LOCDA,IV,IH)                                             TEC   11
      LOC=LOCDA+1                                                       TEC   12
      CALL UN2(LOC,IN,IA)                                               TEC   13
      LOC=LOC+1                                                         TEC   14
      CALL UN2(LOC,IR1,IR2)                                             TEC   15
      IRC=MASTER(LOC+1)                                                 TEC   16
      R1=ASTER(IR1)                                                     TEC   17
      R2=ASTER(IR2)                                                     TEC   18
      R3=R1/ASTER(IRC)                                                  TEC   19
      R4=R2/ASTER(IRC)                                                  TEC   20
      DDN=0.                                                            TEC   21
      WDA=0.                                                            TEC   22
      DDA=0.                                                            TEC   23
      HDA=0.                                                            TEC   24
      HDN=0.                                                            TEC   25
```

```
      WDN=0.                                                            TEC   26
      DO 100 I=1,3                                                      TEC   27
      I1=I-1                                                            TEC   28
      J1=IV+I1                                                          TEC   29
      J2=IH+I1                                                          TEC   30
      J3=IN+I1                                                          TEC   31
      J4=IA+I1                                                          TEC   32
      DELTA(I)=ASTER(J1)-XB(I)                                          TEC   33
      HF(I)=ASTER(J2)                                                   TEC   34
      NF(I)=ASTER(J3)                                                   TEC   35
      AUN(I)=ASTER(J4)                                                  TEC   36
      DDN=DELTA(I)*NF(I)+DDN                                            TEC   37
      WDA=WB(I)*AUN(I)+WDA                                              TEC   38
      DDA=DELTA(I)*AUN(I)+DDA                                           TEC   39
      HDA=HF(I)*AUN(I)+HDA                                              TEC   40
      HDN=HF(I)*NF(I)+HDN                                               TEC   41
      WDN=WB(I)*NF(I)+WDN                                               TEC   42
  100 CONTINUE                                                          TEC   43
      CALL CROSS(K,AUN,NF)                                              TEC   44
      WDK=DOT(WB,K)                                                     TEC   45
      DDK=DOT(DELTA,K)                                                  TEC   46
      HDK=DOT(HF,K)                                                     TEC   47
      IF(ABS(WDN).GT.1.0E-7)GOTO 300                                    TEC   48
      GAMMA=-DDN/HDN                                                    TEC   49
      IF(GAMMA.LT.0.)GOTO 900                                           TEC   50
      RTP=GAMMA-1.                                                      TEC   51
      IF(RTP.GT.0.)GOTO 900                                             TEC   52
      M=GAMMA*R3+R1*(1.-GAMMA)                                          TEC   53
      MM=GAMMA*R4+R2*(1.-GAMMA)                                         TEC   54
      M2=M*M                                                            TEC   55
      MM2=MM*MM                                                         TEC   56
      T=SIGMA*HDA+DDA                                                   TEC   57
      TT=SIGMA*HDK+DDK                                                  TEC   58
      A=MM2*WDA**2+M2*WDK**2                                            TEC   59
      B=-(MM2*WDA*T+M2*WDK*TT)                                          TEC   60
      C=MM2*T**2+M2*TT**2-M2*MM2                                        TEC   61
      DISC=B*B-A*C                                                      TEC   62
      IF(DISC.LT.0.)GOTO 900                                            TEC   63
      IF(DISC.GT.0.)DISC=SQRT(DISC)                                     TEC   64
      RIN=(-B-DISC)/A                                                   TEC   65
      ROUT=(-B+DISC)/A                                                  TEC   66
      LRI=3                                                             TEC   67
      LRO=3                                                             TEC   68
      GOTO 950                                                          TEC   69
C                                                                       TEC   70
  300 FLIPD=1.                                                          TEC   71
      IF(WDN.LT.0.)GOTO 310                                             TEC   72
      FLIPD=-1.                                                         TEC   73
      WDA=-WDA                                                          TEC   74
      WDN=-WDN                                                          TEC   75
      WDK=-WDK                                                          TEC   76
  310 ALPHA=HDN/WDN                                                     TEC   77
      BETA=DDN/WDN                                                      TEC   78
      TAU=(R3/R4)**2                                                    TEC   79
      A=(ALPHA*WDA-HDA)**2+TAU*(ALPHA*WDK-HDK)**2-TAU*(R4-R2)**2        TEC   80
      B=     -(-ALPHA*BETA*WDA**2+ALPHA*WDA*DDA+BETA*WDA*HDA-DDA*HDA    TEC   81
     1  +TAU*(-ALPHA*BETA*WDK**2+ALPHA*WDK*DDK+BETA*WDK*HDK-DDK*HDK     TEC   82
     2  +R2*R4-R2*R2))                                                  TEC   83
      C=(DDA-BETA*WDA)**2+TAU*((DDK-BETA*WDK)**2-R2**2)                 TEC   84
      DISC=B*B-A*C                                                      TEC   85
```

```
      IF(DISC.LT.0.)GOTO 900
      IF(DISC.GT.0.)DISC=SQRT(DISC)
      IF(ABS(A).LE.1.0E-7)GOTO 330
      IF(A)320,330,340
  320 SIGMA1=(-B-DISC)/A
      SIGMA2=(-B+DISC)/A
      GOTO 350
  330 SIGMA1=-C/(2.*B)
      SIGMA2=-PINF
      IF(SIGMA1)900,350,350
  340 SIGMA1=(-B+DISC)/A
      SIGMA2=(-B-DISC)/A
  350 SIGMAP=-R1/(R3-R1)
      IF(SIGMA2.GT.1.)GOTO 900
      IF(SIGMA1.LT.0.)GOTO 900
      IF(SIGMA1.GT.1.)GOTO 410
      IF(SIGMA2.GT.0.)GOTO 400
      RIN=ALPHA*SIGMA1+BETA
      LRI=3
      ROUT=BETA
      LRO=1
      GOTO 490
  400 RIN=ALPHA*SIGMA1+BETA
      LRI=3
      ROUT=ALPHA*SIGMA2+BETA
      LRO=3
      GOTO 490
  410 IF(SIGMA2.GT.0.)GOTO 440
      IF(SIGMA1.GT.SIGMAP)GOTO 900
      RIN=ALPHA+BETA
      LRI=2
      ROUT=BETA
      LRO=1
      GOTO 490
  440 IF(SIGMA1.GT.SIGMAP)GOTO 460
      RIN=ALPHA+BETA
      LRI=2
      ROUT=ALPHA*SIGMA2+BETA
      LRO=3
      GOTO 490
  460 RIN=ALPHA*SIGMA2+BETA
      LRI=3
      ROUT=BETA
      LRO=1
C
  490 IF(FLIPD.GE.0.)GOTO 950
      RTP=RIN
      ITP=LRI
      RIN=-ROUT
      LRI=LRO
      ROUT=-RTP
      LRO=ITP
      GOTO 950
  900 RIN=PINF
      ROUT=-PINF
  950 IF(ROUT.GT.0.)GOTO 1000
      RIN=PINF
      ROUT=-PINF
      RETURN
 1000 IF(ABS(ROUT-RIN).LE.RIN*1.0E-6)GOTO 900
```

```
      RETURN                                                            TEC  146
      END                                                               TEC  147
C                                                                       TEC  148
C                                                                       TEC  149
      SUBROUTINE TOR                                                    ****  40
      DIMENSION MASTER(30000),XMCV(3),C(4),RT(4),RTS(4),XAW(3),XTRY(3)  TOR    2
      COMMON ASTER(30000)                                               TOR    3
      COMMON/PAREM/XB(3),WB(3),IR                                       TOR    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 TOR    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     TOR    6
     1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP              TOR    7
      EQUIVALENCE (MASTER,ASTER),(DIST,STHUS)                           TOR    8
      REAL NF(3)                                                        TOR    9
C                                                                       TOR   10
      CALL UN2(LOCDA,IV,IN)                                             TOR   11
      LOC=LOCDA+1                                                       TOR   12
      CALL UN2(LOC,IR1,IR2)                                             TOR   13
      R1=ASTER(IR1)                                                     TOR   14
      R2=ASTER(IR2)                                                     TOR   15
      WDN=0.                                                            TOR   16
      XMC2=0.                                                           TOR   17
      AW=SQRT(DOT(WB,WB))                                               TOR   18
      DO 10 I=1,3                                                       TOR   19
      J1=IV+I-1                                                         TOR   20
      XAW(I)=ASTER(J1)-XB(I)                                            TOR   21
   10 CONTINUE                                                          TOR   22
      RSAVE=ABS(DOT(XAW,WB)/AW)-R1-R2-R2                                TOR   23
      IF(NASC.EQ.-2)RSAVE=0.                                            TOR   24
      DO 20 I=1,3                                                       TOR   25
      XTRY(I)=XB(I)+RSAVE*WB(I)                                         TOR   26
   20 CONTINUE                                                          TOR   27
      DO 100 I=1,3                                                      TOR   28
      J1=IV+I-1                                                         TOR   29
      J2=IN+I-1                                                         TOR   30
      NF(I)=ASTER(J2)                                                   TOR   31
      XMCV(I)=XTRY(I)-ASTER(J1)                                         TOR   32
      XMC2=XMCV(I)**2+XMC2                                              TOR   33
      WDN=WB(I)*ASTER(J2)+WDN                                           TOR   34
  100 CONTINUE                                                          TOR   35
      WDXMC=DOT(WB,XMCV)                                                TOR   36
      XMCDN=DOT(XMCV,NF)                                                TOR   37
      R12=R1*R1                                                         TOR   38
      R22=R2*R2                                                         TOR   39
      TERM=R12+R22-XMC2                                                 TOR   40
      C(1)=4.*WDXMC                                                     TOR   41
      TEMP=4.*WDXMC**2                                                  TOR   42
      C(2)=4.*R12*WDN**2-2.*TERM+TEMP                                   TOR   43
      C(3)=8.*R12*WDN*XMCDN-4.*WDXMC*TERM                               TOR   44
      C(4)=4.*R12*(XMCDN**2-R22)+TERM**2                                TOR   45
      CALL QRTIC(C,RT,NR)                                               TOR   46
      IF(NR-2)110,120,140                                               TOR   47
C         TOR NOT HIT                                                   TOR   48
  110 RIN=0.                                                            TOR   49
      ROUT=-PINF                                                        TOR   50
      RETURN                                                            TOR   51
C         2 ROOTS                                                       TOR   52
  120 IF(RT(1).GE.RT(2))GOTO 130                                        TOR   53
      RIN=RT(1)                                                         TOR   54
      ROUT=RT(2)                                                        TOR   55
      GOTO 900                                                          TOR   56
```

```
  130 RIN=RT(2)                                                         TOR   57
      ROUT=RT(1)                                                        TOR   58
      GOTO 900                                                          TOR   59
C         4 ROOTS    SELECT FIRST PAIR .GE. DIST AS RIN AND ROUT        TOR   60
  140 RTS(1)=RT(1)                                                      TOR   61
      IF(RT(2).LT.RTS(1))GOTO 150                                       TOR   62
      RTS(2)=RT(2)                                                      TOR   63
      GOTO 160                                                          TOR   64
  150 RTS(2)=RTS(1)                                                     TOR   65
      RTS(1)=RT(2)                                                      TOR   66
  160 IF(RT(3).LT.RTS(2))GOTO 170                                       TOR   67
      RTS(3)=RT(3)                                                      TOR   68
      GOTO 190                                                          TOR   69
  170 RTS(3)=RTS(2)                                                     TOR   70
      IF(RT(3).LT.RTS(1))GOTO 180                                       TOR   71
      RTS(2)=RT(3)                                                      TOR   72
      GOTO 190                                                          TOR   73
  180 RTS(2)=RTS(1)                                                     TOR   74
      RTS(1)=RT(3)                                                      TOR   75
  190 IF(RT(4).LT.RTS(3))GOTO 200                                       TOR   76
      RTS(4)=RT(4)                                                      TOR   77
      GOTO 300                                                          TOR   78
  200 RTS(4)=RTS(3)                                                     TOR   79
      IF(RT(4).LT.RTS(2))GOTO 210                                       TOR   80
      RTS(3)=RT(4)                                                      TOR   81
      GOTO 300                                                          TOR   82
  210 RTS(3)=RTS(2)                                                     TOR   83
      IF(RT(4).LT.RTS(1))GOTO 220                                       TOR   84
      RTS(2)=RT(4)                                                      TOR   85
      GOTO 300                                                          TOR   86
  220 RTS(2)=RTS(1)                                                     TOR   87
      RTS(1)=RT(4)                                                      TOR   88
C         STHUS=DIST                                                    TOR   89
  300 IF(ABS(STHUS-RTS(2)).LE.1.0E-7)GOTO 310                           TOR   90
      IF(STHUS.GE.RTS(2))GOTO 310                                       TOR   91
      RIN=RTS(1)                                                        TOR   92
      ROUT=RTS(2)                                                       TOR   93
      GOTO 900                                                          TOR   94
  310 RIN=RTS(3)                                                        TOR   95
      ROUT=RTS(4)                                                       TOR   96
C                                                                       TOR   97
  900 LRI=1                                                             TOR   98
      LRO=1                                                             TOR   99
      RIN=RIN+RSAVE                                                     TOR  100
      ROUT=ROUT+RSAVE                                                   TOR  101
      IF(ROUT.GE.0.0)GOTO 920                                           TOR  102
  910 RIN=PINF                                                          TOR  103
      ROUT=-PINF                                                        TOR  104
      RETURN                                                            TOR  105
  920 IF(ABS(ROUT-RIN).LE.RIN*1.0E-6)GOTO 910                           TOR  106
      RETURN                                                            TOR  107
      END                                                               TOR  108
C                                                                       TOR  109
C                                                                       TOR  110
      SUBROUTINE TRC                                                    ****  41
      DIMENSION MASTER(30000),V(3),H(3)                                 TRC    2
      COMMON ASTER(30000)                                               TRC    3
      COMMON/PAREM/XB(3),WB(3),IR                                       TRC    4
      COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST                 TRC    5
      COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,     TRC    6
```

```
      1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
       EQUIVALENCE(MASTER,ASTER)
C
       CALL UN2(LOCDA,IV,IH)
       LOC=LOCDA+1
       CALL UN2(LOC,IRB,IRTOP)
       V(1)=ASTER(IV)
       V(2)=ASTER(IV+1)
       V(3)=ASTER(IV+2)
       H(1)=ASTER(IH)
       H(2)=ASTER(IH+1)
       H(3)=ASTER(IH+2)
       RB=ASTER(IRB)
       RT=ASTER(IRTOP)
       RIN=-PINF
       ROUT=PINF
       LRO=0
       LRI=0
       INTSEC=0
       INTR1=0
       INTR2=0
       V1XB1=V(1)-XB(1)
       V2XB2=V(2)-XB(2)
       V3XB3=V(3)-XB(3)
       PVPV=V1XB1*V1XB1+V2XB2*V2XB2+V3XB3*V3XB3
       VPW=V1XB1*WB(1)+V2XB2*WB(2)+V3XB3*WB(3)
       WH =WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)
       VPH=V1XB1*H(1)+V2XB2*H(2)+V3XB3*H(3)
       HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)
       RTRB=RT-RB
       RBRTVP=RB-VPH*RTRB/HH
       VPHHH=VPH+HH
       UM=HH*(PVPV-RBRTVP**2)-VPH*VPH
       AMBD=HH*VPW-WH*(VPH-RTRB*RBRTVP)
       DEN=HH-WH**2*(1.+RTRB**2/HH)
       IF(ABS(DEN).GT.1.0E-6)GOTO 40
       IF(RTRB.EQ.0.)GOTO 200
       R2=UM/(2.*AMBD)
       F1=R2*WH-VPH
       IF(F1.LT.0.)GOTO 200
       IF(F1.GT.HH)GOTO 200
       INTSEC=INTSEC+1
       IF(WH.LE.0.)GOTO 10
       IF(RTRB)20,20,30
    10 IF(RTRB)30,30,20
    20 LRO=3
       ROUT=R2
       GOTO 250
    30 LRI=3
       RIN=R2
       INTSEC=INTSEC+1
       GOTO 210
C
    40 AMBDA=AMBD/DEN
       UMU=UM/DEN
       DISC=AMBDA**2-UMU
       IF(DISC)350,200,50
       GOTO 50
    50 SD=SQRT(DISC)
       R1=AMBDA-SD
```

```
      R2=AMBDA+SD                                                       TRC   67
      F1=R2*WH-VPH                                                      TRC   68
      IF(F1.LT.0.)GOTO 60                                               TRC   69
      IF(F1.LE.HH)INTR2=INTR2+1                                         TRC   70
   60 F1=R1*WH-VPH                                                      TRC   71
      IF(F1.LT.0.)GOTO 70                                               TRC   72
      IF(F1.LE.HH)GOTO 80                                               TRC   73
   70 IF(INTR2.LT.1)GOTO 200                                            TRC   74
      ROUT=R2                                                           TRC   75
      RIN=R2                                                            TRC   76
      LRO=3                                                             TRC   77
      LRI=3                                                             TRC   78
      INTSEC=INTSEC+1                                                   TRC   79
      GOTO 200                                                          TRC   80
   80 INTR1=INTR1+1                                                     TRC   81
      IF(INTR2.GE.1)GOTO 90                                             TRC   82
      ROUT=R1                                                           TRC   83
      RIN=R1                                                            TRC   84
      LRO=3                                                             TRC   85
      LRI=3                                                             TRC   86
      INTSEC=INTSEC+1                                                   TRC   87
      GOTO 200                                                          TRC   88
   90 IF(R1-R2)100,350,110                                              TRC   89
  100 RIN=R1                                                            TRC   90
      ROUT=R2                                                           TRC   91
      LRO=3                                                             TRC   92
      LRI=3                                                             TRC   93
      GOTO 300                                                          TRC   94
  110 RIN=R2                                                            TRC   95
      ROUT=R1                                                           TRC   96
      LRO=3                                                             TRC   97
      LRI=3                                                             TRC   98
      GOTO 300                                                          TRC   99
C                                                                       TRC  100
  200 IF(WH)210,350,250                                                 TRC  101
  210 IF(VPH.GE.0.)GOTO 350                                             TRC  102
      CP=VPH/WH                                                         TRC  103
      F1=CP*CP-2.*CP*VPW+PVPV-RB*RB                                     TRC  104
      IF(F1.GT.0.)GOTO 220                                              TRC  105
      INTSEC=INTSEC+1                                                   TRC  106
      ROUT=CP                                                           TRC  107
      LRO=1                                                             TRC  108
      IF(INTSEC.GE.2)GOTO 300                                           TRC  109
  220 CM=VPHHH/WH                                                       TRC  110
      F1=CM*CM-2.*((VPW+WH)*CM-VPH)+HH+PVPV-RT*RT                       TRC  111
      IF(F1.GT.0.)GOTO 350                                              TRC  112
      RIN=CM                                                            TRC  113
      LRI=2                                                             TRC  114
      GOTO 300                                                          TRC  115
  250 IF(VPHHH.LT.0.)GOTO 350                                           TRC  116
      CP=VPHHH/WH                                                       TRC  117
      F1=CP*CP-2.*((VPW+WH)*CP-VPH)+HH+PVPV-RT*RT                       TRC  118
      IF(F1.GT.0.)GOTO 260                                              TRC  119
      INTSEC=INTSEC+1                                                   TRC  120
      ROUT=CP                                                           TRC  121
      LRO=2                                                             TRC  122
  260 IF(INTSEC.GE.2)GOTO 300                                           TRC  123
      CM=VPH/WH                                                         TRC  124
      F1=CM*CM-2.*CM*VPW+PVPV-RB*RB                                     TRC  125
      IF(F1.GT.0.)GOTO 350                                              TRC  126
```

```
      RIN=CM
      LRI=1
C
  300 IF(ABS(ROUT-RIN)-ROUT*1.0E-5)350,350,360
  350 RIN=PINF
      ROUT=-PINF
      LRI=0
      LRO=0
  360 RETURN
      END
C
C
END
```


<<